MAR 2 2 1988

126 D-3

C60.10:88-230

NTIA Report 88-230

C60.10:88-230

Use of Precalculated Data of Minimum Elliptical Beams of Satellite Antennas

Hiroshi Akima

COMPLETED

report series

U.S. DEPARTMENT OF COMMERCE • National Telecommunications and Information Administration

NTIA Report 88-230

Use of Precalculated Data of Minimum Elliptical Beams of Satellite Antennas

Hiroshi Akima

U.S. DEPARTMENT OF COMMERCE
C. William Verity, Secretary
Alfred C. Sikes, Assistant Secretary
for Communications and Information

February 1988

0
1

BLANK PAGE

CONTENTS

	Page
ABSTRACT	1
1. INTRODUCTION	1
2. ANALYSIS OF THE FIRST METHOD	4
3. ANALYSIS OF THE SECOND METHOD	7
4. SOFTWARE DEVELOPMENT	11
5. TIMING ANALYSIS	14
6. CONCLUSIONS	18
7. RECOMMENDATIONS	19
8. ACKNOWLEDGMENTS	19
9. REFERENCES	20
APPENDIX A. THE CDMNEB PROGRAM	21
APPENDIX B. THE MNEBSA SUBROUTINE PACKAGE	41
APPENDIX C. THE MNEBIP SUBROUTINE	49

USE OF PRECALCULATED DATA OF MINIMUM ELLIPTICAL BEAMS OF SATELLITE ANTENNAS

Hiroshi Akima[1]

Minimum (or "best-fit") elliptical beams of satellite antennas have been used for the planning of space services utilizing the geostationary-satellite orbit. In such a planning, beam parameters of minimum elliptical beams are necessary for all possible orbital positions of all satellites. Since calculation of beam parameters of every minimum elliptical beam may be rather time-consuming, an alternative method is desirable. This study shows that the beam parameters for a satellite longitude position calculated by linear interpolation from the beam parameters for two nearby longitude positions can be used without a serious degradation of antenna gain. This report describes a method in which the beam parameters for minimum elliptical beams are precalculated for a sufficient number of satellite longitude positions and the beam parameters for a desired longitude position are calculated from the beam parameters for two nearby longitude positions that bound the longitude position in question. Fortran programs that implement this method also are presented.

Key words: antenna beam; elliptical beam; linear interpolation; minimum elliptical beam; satellite antenna

1. INTRODUCTION

An elliptical beam of a satellite antenna is one of the most common beam shapes used for a communication satellite in geostationary-satellite orbit. Minimizing the area of the ellipse (or solid angle) of the beam (or best-fitting the beam to the service area) is desirable for saving the transmitter power and suppressing harmful interference to other communication networks. In the ITU (International Telecommunication Union), a minimum elliptical beam of a satellite antenna is specified with five parameters, i.e., the longitude and latitude of the aim point (or boresight) on the surface of the Earth, the

---

[1]The author is with the Institute for Telecommunication Sciences, National Telecommunications and Information Administration, U.S. Department of Commerce, Boulder, Colorado 80303-3328.

major-axis and minor-axis beamwidths of the ellipse, and the orientation angle of the major axis of the ellipse (ITU, 1986). (The orientation angle of the ellipse is defined, in a plane normal to the beam axis, as the angle measured counterclockwise from a line parallel to the equatorial plane to the major axis of the ellipse.) An algorithm to determine the five parameters of a minimum elliptical beam of a satellite antenna exists (Akima, 1981), but determining a large number of minimum elliptical beams with the algorithm may be rather time consuming.

The ITU has been planning space services utilizing the geostationary-satellite orbit,[2] and a minimum elliptical beam of a satellite antenna has been used for each service area in the planning. Since many satellites are to be placed in the geostationary-satellite orbit in the planning, use of the orbital arc must be optimized, i.e., the longitude positions of the satellites must be adjusted in such a way that the signal-to-interference ratios for all satellite networks are above a specified threshold value and the total orbital arc used is reasonably small. In such an optimization process, beam parameters of minimum elliptical beams are necessary for all possible orbital positions of all satellites. Since the on-line calculation of these beam parameters may be time-consuming, a method that obviates the on-line calculation of an elliptical beam each time is desirable. We consider two candidate methods for this purpose.

It is conceivable that the data calculated for a service area and for a longitude position of the satellite may be valid if the satellite is moved on the geostationary-satellite orbit by only a small amount. The first method uses, as the beam data for a specified satellite longitude position, the precalculated beam data for the closest satellite longitude position. Then, the question is, How much can the satellite be moved with the same elliptical beam data without a serious loss in the antenna gain tolerance? (The tolerance in antenna gain is defined here as the excess of antenna gain above the half-power (or -3 dB) gain level and is calculated on the assumption that the antenna gain expressed in decibels varies as the minus of the square of the

---

[2]The ITU already planned the Broadcasting-Satellite Service in the 12-GHz band (ITU, 1977, 1983, 1986), and it will plan the Fixed-Satellite Service in the so-called extension band of the 6 and 4-GHz bands in the second session of the WARC-ORB (World Administrative Radio Conference on the Use of the Geostationary-Satellite Orbit and the Planning of the Space Services Utilizing It) to be convened in 1988.

off-axis angle around the half-power point.) This report examines this problem in Section 2.

Another way of using precalculated data is to interpolate linearly the five parameters of the elliptical beam between the data for two nearby satellite longitude positions that bound the longitude position in question. The second method implements this idea. Then, the question is, How much spacing is allowed between a pair of satellite longitude positions for the interpolated data not to cause a serious loss in the antenna gain tolerance? This problem is examined in Section 3.

As may be expected, it turns out that the second method based on linear interpolation is superior to the first method based on the beam data for the closest position. The second method requires fewer satellite longitude positions for which minimum elliptical beams have to be precalculated. Another advantage of the second method is that the five parameters determined by the second method are continuous functions of the desired satellite longitude position while the parameters determined by the first method are discontinuous functions. We have, therefore, implemented the second method in computer software. The developed software is described in Section 4.

We have analyzed the calculation time required for the developed software. As expected, the calculation time required for determining an elliptical beam with the developed software compares very favorably with the calculation time required for calculating the minimum elliptical beam from scratch. The results of calculation time analysis performed on the developed software are presented in Section 5.

Concluding remarks are given in Section 6, and recommendations based on this study are given in Section 7.

Fortran codes of the computer software developed in this study are listed with pertinent descriptions in three appendices to this report. The MNEBSA subroutine package that determines the minimum elliptical beam and calculates the antenna gain tolerance is listed in Appendix A. The CDMNEB program that calculates the necessary beam parameters by calling the MNEBSA subroutine and stores the calculated beam parameter values in a data file is listed in Appendix B. Finally, the MNEBIP subroutine that calculates the beam parameters by interpolating the values in the data file created by the CDMNEB program is listed in Appendix C.

2. ANALYSIS OF THE FIRST METHOD

To determine the elliptical beam of a satellite antenna for a given satellite longitude position in the geostationary-satellite orbit and for a given service area, this method uses precalculated data of the minimum elliptical beam for the closest satellite longitude position.

To analyze the method, we have first determined the minimum elliptical beam for every degree of visible satellite longitude position (i.e., the position above the horizon) and for each of 30 service areas, including 12 Region-2 service areas, 15 Region-3 service areas, and 3 service areas artificially drawn for test purposes.[3] For each beam, we have calculated the tolerance in antenna gain of the elliptical beam that has the same beam parameters but has been displaced (or offset) in the geostationary-satellite orbit from the original satellite longitude position. As defined earlier, the tolerance in antenna gain is the excess of antenna gain above the half-power (or -3 dB) gain level and is calculated on the assumption that the antenna gain expressed in decibels varies as the minus of the square of the off-axis angle around the half-power point. We have calculated the tolerances for five values of orbital displacement, i.e., 0.25°, 0.5°, 1.0°, 2.0°, and 3.0°, for every degree of original satellite longitude position, and for the 30 service areas.

To determine the minimum elliptical beams and to calculate the antenna gain tolerances in this analysis, we have used a modified version of the MNEBSA subroutine package. Although slightly modified, this subroutine package is essentially the same as its original version developed by Akima (1981). We have called this subroutine package with the following input values, i.e., the minimum elevation angle of 0.0°, minimum beamwidth of 1.0°, pointing error of 0.1°, and orientation-angle error of 1.0°. (See Appendix A.)

The calculation results have indicated that the tolerance also depends on the elevation angle of the satellite for the service area (which is defined as the minimum of the elevation angles of the satellite seen from all polygon points in the service area and is also the elevation angle of the satellite seen from the farthest polygon point in the service area) as well as the displacement of the satellite just described. Therefore, we have accumulated all tolerance values for each degree of elevation angle and for each

---

[3]The ITU Region 2 includes North America, Central America, South America, and Greenland. Region 3 includes Asia and Oceania.

displacement value without regard to the satellite longitude positions and service areas, and we have selected the minimum of the tolerance values accumulated in each bin. Figure 1 shows the minimum antenna gain tolerance, parametric in the displacement, as a function of elevation angle. The figure indicates that, as expected, the tolerance decreases as the displacement increases. The figure also indicates that the minimum value of the tolerance in antenna gain is a monotonic increasing function of the elevation angle of the satellite for the service area. If the minimum (i.e., worst) tolerance is specified at -0.1 dB, the minimum elevation angles required are about 0°, 20°, 40°, 60°, and 70° for the displacements of 0.25°, 0.5°, 1.0°, 2.0°, and 3.0°, respectively. When the minimum tolerance is -0.1 dB, therefore, it would be sufficient to precalculate the minimum elliptical beam at every 0.5° interval for the portion of the orbit that yields an elevation angle of 20° or less, at every 1° interval for an elevation angle between 20° and 40°, at every 2° interval for an elevation angle between 40° and 60°, at every 4° interval for an elevation angle between 60° and 70°, and at every 6° interval elsewhere.

If the minimum tolerance is relaxed to -0.2 dB, the minimum elevation angles required are about 0°, 0°, 20°, 40°, and 55° for the same displacement values. When the minimum tolerance is -0.2 dB, therefore, it would be sufficient to precalculate the minimum elliptical beam at every 1° interval for the portion of the orbit that yields an elevation angle of 20° or less, at every 2° interval for an elevation angle between 20° and 40°, at every 4° interval for an elevation angle between 40° and 55°, and at every 6° interval elsewhere.

Similarly, when the minimum tolerance is -0.3 dB, it would be sufficient to precalculate the minimum elliptical beam at every 1° interval for the portion of the orbit that yields an elevation angle of 10° or less, at every 2° interval for an elevation angle between 10° and 30°, at every 4° interval for an elevation angle between 30° and 45°, and at every 6° interval elsewhere.

When the minimum tolerance is specified at -0.1 dB, a maximum of 145 satellite longitude positions per service area covers the entire visible orbital arc. (The maximum number of satellite longitude positions is required for a very small service area located on the Equator.) The size of the data file to store the beams data is 6 x 145 = 870 words. (Six words for a beam include the satellite longitude position and five elliptical beam parameters.) If 4 bytes per word are assumed, the file size for a service area is

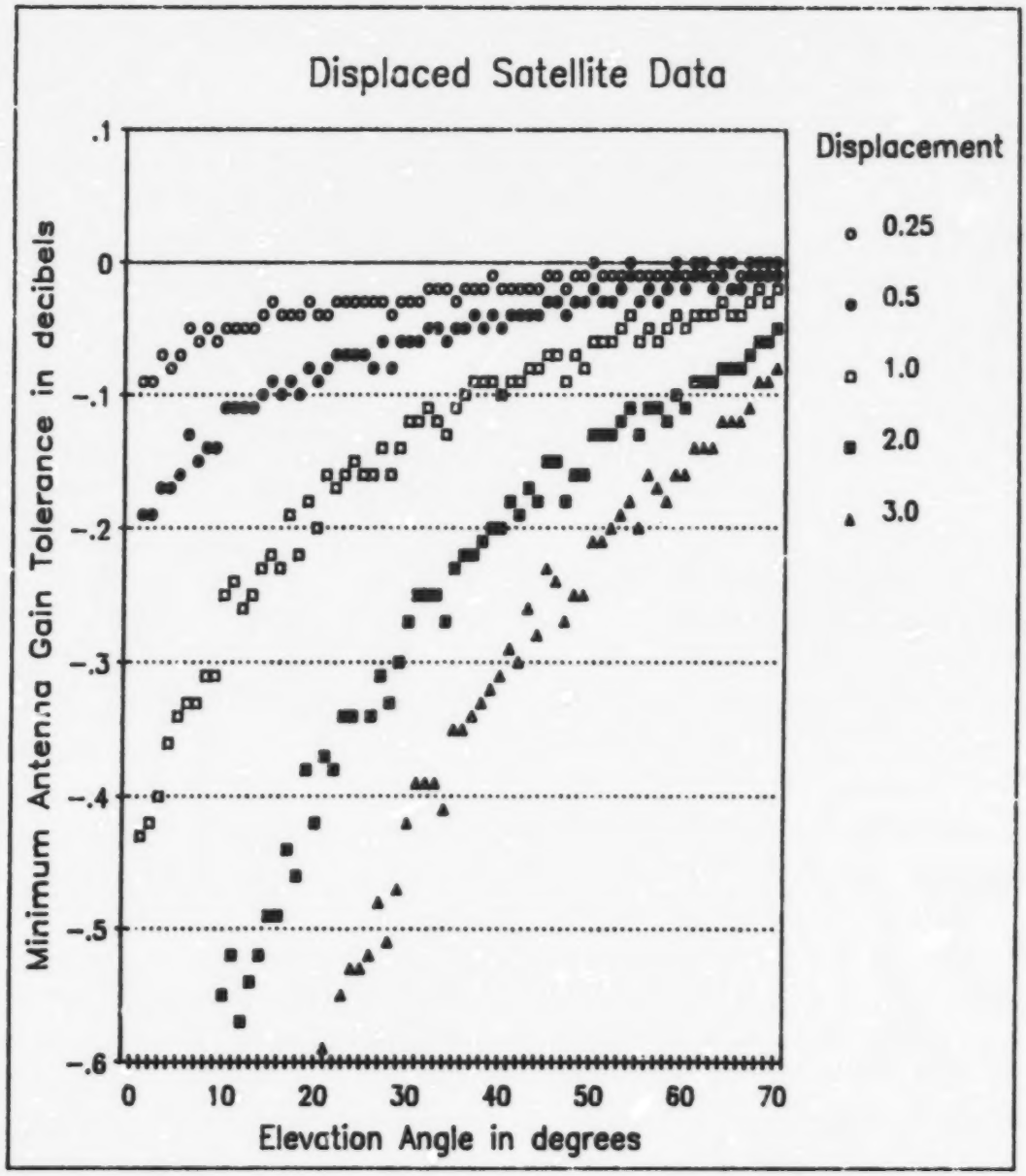


Figure 1. Minimum antenna gain tolerance as a function of elevation angle. (Satellites are displaced from their original longitude positions.)

BEST COPY AVAILABLE

approximately 3.4 kilobytes, where we follow a convention that 1 kilobyte equals 1,024 bytes. If we assume that 300 service areas in the world are to be considered in the 1988 WARC-ORB (World Administrative Radio Conference on the Use of the Geostationary-Satellite Orbit and the Planning of Space Services Utilizing It), the total size of the data file is 870 x 300 = 261,000 words, or approximately 1,020 kilobytes.

When the minimum tolerance is relaxed to -0.2 dB, a maximum of 79 satellite longitude positions covers the entire visible orbital arc. The size of the data file to store the beams data per service area is approximately 1.85 kilobyte, and the total size of the data file for 300 service areas is approximately 555 kilobytes.

Similarly, when the minimum tolerance is relaxed to -0.3 dB, a maximum of 64 satellite longitude positions covers the entire visible orbital arc. The size of the data file to store the beams data per service area is approximately 1.5 kilobyte, and the total size of the data file for 300 service areas is approximately 450 kilobytes.

If a uniform spacing of 0.5° is used without optimizing the spacing, the minimum tolerance is -0.1 dB for the entire visible orbital arc, and the total file size for 300 service areas is approximately 2300 kilobytes. (Note that the width of the visible arc for a service area never exceeds 162.6°.) If a uniform spacing of 1° is used, the minimum tolerance is -0.2 dB for the entire visible arc, and the total file size is approximately 1150 kilobytes.

The time required to retrieve the beam data from the data file of this size is not expected to be too excessive. In this method, no additional time is required to determine the elliptical beam from the retrieved data. (See Section 5.)

3. ANALYSIS OF THE SECOND METHOD

To determine the elliptical beam of a satellite antenna for a given satellite longitude position and for a given service area, this method uses two sets of precalculated data, i.e., the minimum elliptical beams for two satellite longitude positions that bound the satellite longitude position in question. The method linearly interpolates each beam parameter between the two sets of data.

To analyze the method, we have first determined the minimum elliptical beam for every degree of visible satellite longitude position (i.e., the position above the horizon) and for each of the same 30 service areas as used in the analysis of the first method. For satellite longitude positions at 0.5° intervals, we have interpolated each beam parameter between the two values for the minimum elliptical beams for the two satellite longitude positions that bound the satellite longitude position in question, and we have calculated the antenna gain tolerance of the elliptical beam thus interpolated. We have calculated the tolerances for three values of the interval between the two satellite longitude positions, i.e., 4.0°, 6.0°, and 8.0°, for satellite longitude positions at 0.5° intervals, and for the 30 service areas.

To determine the minimum elliptical beams and to calculate the antenna gain tolerances in this analysis, we have used the modified version of the MNEBSA subroutine package with the same values of input parameters as in the analysis of the first method.

We have accumulated all tolerance values for each degree of elevation angle and for each interval value without regard to the satellite longitude positions and service areas, and we have selected the minimum of the tolerance values accumulated in each bin. Figure 2 shows the minimum antenna gain tolerance, parametric in the interval, as a function of elevation angle. The figure indicates that the tolerance is equal to -0.1 dB or greater if the interval between the two positions is 4°, that the tolerance is greater than -0.2 dB if the interval between the two positions is 6° or less, and that the tolerance is greater than -0.3 dB if the interval between the two positions is 8° or less. If the minimum (i.e., worst) tolerance is specified at -0.1 dB, therefore, it is sufficient to precalculate the minimum elliptical beam at every 4° interval. For the minimum tolerance of -0.2 dB, precalculations at every 6° interval are sufficient; and for the minimum tolerance of -0.3 dB, at every 8° interval.

When the minimum tolerance is specified at -0.1 dB, a maximum of 42 satellite longitude positions per service area covers the entire visible orbital arc for the service area. (Note that the width of the visible arc for a service area never exceeds 162.6°.) If we assume that the elliptical beams data are to be precalculated for 42 satellite longitude positions per service area for an elevation angle of 0° or greater, the size of the data file to store the beams data is calculated as 6 x 42 = 252 words, or approximately

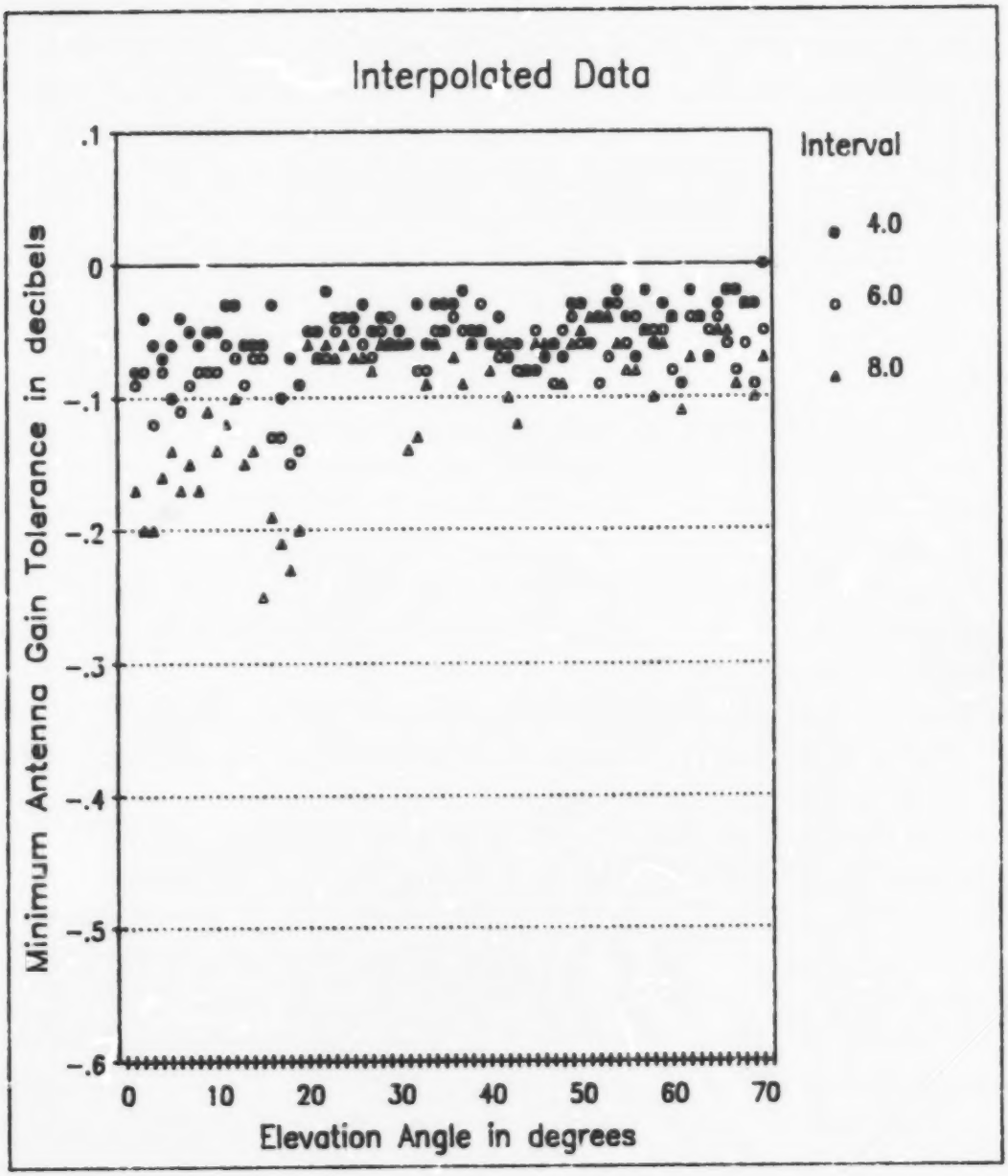


Figure 2. Minimum antenna gain tolerance as a function of elevation angle. (Beam parameters are linearly interpolated between the parameter values for two satellite longitude positions.)

BEST COPY AVAILABLE

1 kilobyte, if 4 bytes per word is assumed. If we assume that 300 service areas are to be considered in the 1988 WARC-ORB, the total size of the data file is calculated as 252 x 300 = 75,600 words, or approximately 300 kilobytes.

When the minimum tolerance is relaxed to -0.2 dB, a maximum of 29 satellite longitude positions covers the entire visible orbital arc. The size of the data file to store the beams data per service area is approximately 700 bytes, and the total size of the data file for 300 service areas is approximately 204 kilobytes.

Similarly, when the minimum tolerance is relaxed to -0.3 dB, a maximum of 22 satellite longitude positions covers the entire visible orbital arc. The size of the data file to store the beams data per service area is approximately 530 bytes, and the total size of the data file for 300 service areas is approximately 155 kilobytes.

We compare, in Table 1, these total data file sizes for the second method with those for the first method. Comparisons of the first and second rows with the third and fourth rows, respectively, indicate that the first method with a uniform spacing requires a much greater file size than the same method with optimized spacing for the same combination of the minimum tolerance and minimum elevation angle. A comparison between the third and sixth rows indicates that the second method requires a much smaller file size than the first method for

Table 1. Comparison of the Required Data File Size
(For entire visible arc for each service area)

	Method	Spacing	Minimum Tolerance (dB)	Data File Size for 300 Service areas (kilobytes)
1	First Method	0.5°	-0.1 dB	2300
2	First Method	1.0°	-0.2 dB	1150
3	First Method	Optimized	-0.1 dB	1020
4	First Method	Optimized	-0.2 dB	555
5	First Method	Optimized	-0.3 dB	450
6	Second Method	4.0°	-0.1 dB	300
7	Second Method	6.0°	-0.2 dB	204
8	Second Method	8.0°	-0.3 dB	155

the same combination of the minimum tolerance and minimum elevation angle. Another comparison between the fourth (or fifth) and sixth rows indicates that the required data file size for the second method for a better tolerance and a wider orbital arc is smaller than the required data file size for the first method for a worse tolerance and a narrower orbital arc.

In the second method, linear interpolation of each elliptical beam parameter is required to determine the elliptical beam from the retrieved data, but the additional time for the interpolation is insignificant compared with the time required for retrieving the data from the data file. (See Section 5.)

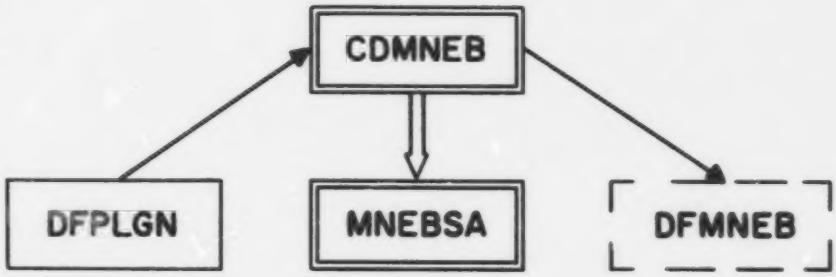
Based on these results, we prefer the second method to the first method. The fact that the second method yields continuous results also supports this choice.

4. SOFTWARE DEVELOPMENT

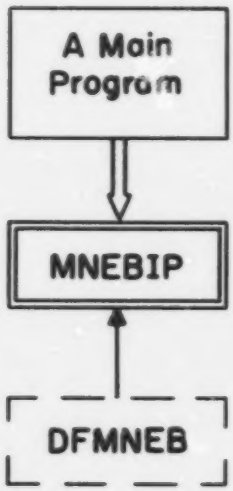
We have written computer software to implement the second method. The software consists of three components, i.e., the CDMNEB program, the MNEBSA subroutine package, and the MNEBIP subroutine. Two data files are associated with this software, i.e., the DFPLGN file and DFMNEB file. The DFPLGN file must be supplied by the user, but the DFMNEB file can be transparent to the user. The functional relationship among these elements is illustrated in Figure 3. Brief descriptions of these components and data files follow.

The CDMNEB Program. This program reads in the polygon-point data of service areas from the DFPLGN data file, determines necessary satellite longitude positions for the service areas, determines minimum elliptical beams for the necessary satellite longitude positions for the service areas, and creates (or appends) the DFMNEB data file that stores the data of the beams. This program calls the MNEBSA subroutine to determine the minimum elliptical beam for a satellite longitude position for a service area. A Fortran listing of the CDMNEB program is given in Appendix B.

The DFPLGN Data File. This data file contains the name and the longitudes and latitudes of the polygon points on the surface of the Earth for each service area. This file must be supplied by the user to run the CDMNEB program. The format of this data file is described in Table 2. An example of the DFPLGN data file is given in Table 3.



(a) Creating or appending the DFMNEB data file.



(b) Use of the DFMNEB data file and MNEBIP subroutine.

A double-line box is provided in this software.
A single-line box is to be provided by the user.
A broken-line box is used inside the software.
A double-line arrow represents a subroutine call.
A single-line arrow represents a data flow.

Figure 3. Relationship among the software elements.

Table 2. The DFPLGN File Format

The first line of the file contains the file name 'DFPLGN' and an optional date in FORMAT (A6,18X,A8). This line is followed by one or more blocks of data for service areas.

The first line of each block of data for a service area contains the following data:

(1) NMSA, i.e., the name of the service area,
(2) NRPP, i.e., the number of polygon points, and
(3) an optional descriptive name of the service area

in FORMAT (A8,8X,I3,5X,A40).

The first line of each block of data is followed by one or more lines, each of which contains up to 10 longitude values of polygon points in FORMAT (10F8.2), followed by one or more lines, each of which contains up to 10 latitude values of polygon points in FORMAT (10F8.2).

Note that the optional date and descriptive name of the service area in this file will not be used when the program is run. They are for bookkeeping purposes only.

Table 3. An Example of the DFPLGN Data File

```
DFPLGN.EXM                87/07/31
ARG          NPP=  6      ARGENTINA
  -52.50  -66.20  -69.00  -73.60  -69.00  -63.20
  -26.30  -21.80  -25.00  -51.00  -55.00  -55.50
CHN          NPP=  6      CHINA (PEOPLE'S REPUBLIC OF)
  130.00   90.00   78.00   90.00  112.00  120.00
   51.00   50.00   35.00   30.00   23.00   37.00
IND          NPP=  6      REPUBLIC OF INDIA
   90.00   80.00   70.00   62.00   75.00   85.00
   25.00   30.00   34.00   25.00    5.00    5.00
MEX          NPP=  5      MEXICO
 -120.00 -105.00  -92.00  -86.00 -105.00
   32.00   18.00   14.00   22.00   32.00
PHL          NPP=  6      REPUBLIC OF THE PHILIPPINES
  125.50  125.80  121.50  120.00  120.00  122.80
    5.50   11.00   19.20   13.80   11.20    7.30
USA/CONU     NPP=  6      USA CONUS (CONTIGUOUS 48 STATES)
 -125.00  -98.00  -82.00  -69.00  -85.00 -125.00
   35.00   26.00   24.00   47.00   49.00   49.00
USA/ET       NPP=  6      USA EASTERN TIME ZONE
  -69.00  -67.00  -75.00  -82.00  -86.00  -91.00
   47.00   45.00   35.00   24.00   30.00   47.00
VEN          NPP=  6      VENEZUELA
  -60.60  -61.00  -72.40  -73.70  -68.00  -65.00
    5.00   11.00   11.70    9.40    0.50    0.50
```

The MNEBSA Subroutine Package. This subroutine package determines the minimum elliptical beam of a satellite antenna for a satellite longitude position and a service area. As mentioned earlier, this package is essentially the same as the original version, developed earlier by Akima (1981). For completeness of this report, a Fortran listing of this subroutine package together with some pertinent descriptions is given in Appendix A.

The DFMNEB Date File. The calculated minimum elliptical beams data are accumulated in this data file when the CDMNEB program is run. This is a direct-access, unformatted file. The structure and contents of this data file are described in the user information of the CDMNEB program, listed in Appendix B. If the user uses the CDMNEB program to create (and append) this data file and uses the MNEBIP subroutine to utilize this data file, this data file is transparent for the user.

The MNEBIP Subroutine. This subroutine calculates the data of elliptical beams of satellite antennas by interpolation. It calculates the data of an elliptical beam of a satellite antenna for a satellite longitude position for each service area by interpolating the data stored in an elliptical beam data file, created or appended by the CDMNEB program. The MNEBIP subroutine opens and closes the DFMNEB data file each time the subroutine is called; the user of this subroutine is not required to open or close the data file. The MNEBIP subroutine is listed in Appendix C.

5. TIMING ANALYSIS

Since the use of precalculated data of minimum elliptical beams of satellite antennas has been proposed to save calculation times, the proposal must be supported by a timing analysis. We have measured the calculation time required for the MNEBIP subroutine that calculates elliptical beams by interpolating the data stored in the DFMNEB data file. For comparison, we have also measured the calculation time required for the MNEBSA subroutine that calculates minimum elliptical beams from scratch.

The same 30 service areas used earlier in the feasibility study have been used for the timing analysis. Two programs, the TMMNEB and TMMNIP programs, that call the MNEBSA and MNEBIP subroutines, respectively, have been written for the analysis. Both programs have been run on the Pyramid 90X computer at

the Institute for Telecommunication Sciences of NTIA, and the "user time" spent in execution of the command and the "system time" spent in the system have been measured by the "time" command of the UNIX system. In this report, we consider the sum of the user and system times as the required time.

The TMMNEB program calculates an elliptical beam per service area for 30 service areas by calling the MNEBSA subroutine. The time required to run the TMMNEB program (for 30 beams) is 330 s. Therefore, the average time required to calculate a minimum elliptical beam by the MNEBSA subroutine is 330/30 = 11.0 s on the Pyramid 90X computer.

The TMMNIP program calculates an elliptical beam per service area for up to 30 service areas by interpolation by calling the MNEBIP subroutine. The time required to run the TMMNIP program depends on the total number of service areas in the data file, N, as well as the number of beams to be calculated, n. The TMMNIP program has been run with seven DFMNEB files. One DFMNEB file, created by the CDMNEB program, contains the data for the N = 30 service areas described earlier. Three files contain the data for N = 100 service areas including the 30 service areas in question in three different ways, i.e., (1) as the first 30 records, (2) distributed uniformly throughout the file, and (3) as the last 30 records. The remaining three files contain the data for N = 300 service areas including the 30 service areas in question in the same three ways as described above. The TMMNIP program has been run for n = 1, 3, and 10 desired beams for n service areas randomly selected from the 30 service areas in question as well as for n = 30 desired beams for all the 30 service areas. Out of the measured times, we have selected, as the required time, the maximum time for each combination of N and n. From the required times for the N and n numbers thus determined, we have calculated the average required times per beam, which are simply the required times for the N and n numbers divided by n. The average required times per beam are shown in Table 4. This table indicates that the average time per beam decreases with n and increases with N. The table also indicates that, even in the worst case for N = 300 and n = 1, the average time per beam is 0.061 s for the MNEBIP subroutine. Even this average time in the worst case compares favorably with 11.0 s, which is the time required for calculation with the MNEBSA subroutine package; MNEBIP is nearly 200 times faster than MNEBSA. If the MNEBIP subroutine is called with n = 10 or greater, MNEBIP is more than 500 times faster than MNEBSA.

Table 4. Average Required Times per Beam for the MNEBIP Subroutine (The second method)

(The n is the number of desired beams, and N is the number of service areas in the data file.)

N	Average time per beam in seconds			
	n = 1	3	10	30
30	0.037	0.019	0.009	0.005
100	0.045	0.023	0.017	0.008
300	0.061	0.029	0.021	0.017

Table 5. Average Required Times per Beam for the Artificial Subroutine (The lower bound for the first method)

(The n is the number of desired beams, and N is the number of service areas in the data file.)

N	Average time per beam in seconds			
	n = 1	3	10	30
30	0.036	0.018	0.008	0.004
100	0.044	0.021	0.015	0.007
300	0.060	0.028	0.020	0.016

The MNEBIP subroutine does the following five steps:

(1) opens the DFMNEB data file,

(2) retrieves minimum elliptical beams data for the desired service area from the DFMNEB file,

(3) locates two satellite longitude positions that bound the desired satellite longitude position,

(4) calculates the five beam parameters by linear interpolation as the output arguments, and

(5) closes the DFMNEB file.

When n is greater than unity, steps (2) through (4) are repeated n times. The fact that steps (1) and (5) are not repeated each time for n = 2 or greater is considered the reason that the average required time per beam decreases as n increases.

If the first method (analyzed in Section 2) were implemented instead of the second method (analyzed in Section 3), the subroutine equivalent to the MNEBIP subroutine would have steps (1), (2), and (5) in common; step (3) would be replaced with a step to locate a satellite longitude position that is closest to the desired satellite longitude position; and step (4) would be replaced with a simple process of replacing the retrieved values for the output arguments. In the subroutine for the first method, step (3) would be a little faster than the MNEBIP subroutine for the second method, and step (4) would require almost no calculation time. Using an artificially written subroutine that skips steps (3) and (4), we have measured the required time for steps (1), (2), and (5). The required time for this artificial subroutine is considered the lower bound for the time required for the subroutine that implements the first method. The results are shown in Table 5 (Page 16) in the same format as in Table 4. Comparison of Table 5 with Table 4 indicates that the time in Table 5 for this artificial subroutine is only 0.001 to 0.002 s faster than the time in Table 4 for MNEBIP for the same combination of N and n. We can, therefore, conclude that time saved resulting from going from the second method to the first method is not enough to justify the use of the first method, which requires a significantly larger data file size and produces discontinuous beam data.

6. CONCLUSIONS

We have studied the feasibility of saving on-line computer time for determining an elliptical beam for a satellite longitude position and for a service area by using precalculated data of minimum elliptical beams of satellite antennas. We have analyzed two candidate methods, i.e., the first method, which uses the precalculated data for a satellite longitude position that is closest to the satellite longitude position in question, and the second method, which interpolates linearly between two sets of data for two positions that bound the satellite longitude position in question. The results of our analyses have indicated that the second method is superior to the first method; the second method requires fewer number of beams to be precalculated, yet produces a better beam for a desired satellite longitude position.

Based on the results, we have developed computer software that implements the second method. Use of the developed computer software consists of two steps, i.e., creation of the beams data file and determination of an elliptical beam for a desired satellite longitude position. In the first step, the CDMNEB program precalculates the minimum elliptical beams by calling the MNEBSA subroutine and creates (and appends) the DFMNEB data file for the precalculated beams data. The first step is an off-line calculation performed independently of the orbit optimization process. The DFMNEB data file is upward compatible in that the user can append the data file by adding beams data for new service areas without disturbing the existing beams data in the data file. In the second step, the MNEBIP subroutine determines an elliptical beam by linearly interpolating the precalculated beams data in the DFMNEB data file. The second step is an on-line calculation in the optimization process. Use of the developed computer software is very simple and straightforward.

The size of the DFMNEB data file created (or appended) in the first step is about 1 kilobyte per service area, if we assume 4 bytes per real number. The computer time required for determining an elliptical beam with the MNEBIP subroutine in the second step is about 200 times faster than that for the calculation of the beam with the MNEBSA subroutine in the worst case, and more than 500 times faster in most practical cases.

7. RECOMMENDATIONS

Based on the studies described in this report, it is recommended

- that data of minimum elliptical beams of satellite antennas be calculated, for every service area, for all satellite longitude positions at 4° intervals in the visible arc of the geostationary-satellite orbit;
- that a data file be set up that stores the data of minimum elliptical beams thus calculated; and
- that, in running an optimization program for the geostationary-satellite orbit, the elliptical beam determined by linear interpolation between the data for two satellite longitude positions that bound the desired satellite longitude position be used without on-line, direct calculation of the minimum elliptical beam.

It also is recommended

- that the CDMNEB program together with the MNEBSA subroutine package be used to calculate the minimum elliptical beams of satellite antennas and set up the data file that stores the calculated data; and
- that the MNEBIP subroutine be used for calculating an elliptical beam of a satellite antenna for a satellite longitude position and for a service area.

8. ACKNOWLEDGMENTS

The author is grateful to Mr. David Netterville of the IFRB (International Frequency Registration Board) of ITU (International Telecommunication Union) for his close examination of the MNEBSA subroutine package with various service areas. He is also grateful to Raymond D. Jennings of the ITS (Institute for Telecommunication Sciences) of NTIA for directing the study, and to Eldon J. Haakinson and Robert F. Kubichek of ITS for their reviews of this report.

9. REFERENCES

Akima, H. (1981), A method for determining the minimum elliptical beam of a satellite antenna, NTIA Report 81-88, October (NTIS Order No. PB82-153966).

ITU (International Telecommunication Union) (1977), Final Acts of the World Administrative Radio Conference for the Planning of the Broadcasting-Satellite Service in Frequency Bands 11.7-12.2 GHz (in Regions 2 and 3) and 11.7-12.5 GHz (in Region 1), ITU, Geneva, Switzerland.

ITU (International Telecommunication Union) (1983), Final Acts of the Regional Administrative Radio Conference for the Planning of the Broadcasting-Satellite Service in Regions 2 (SAT-83), ITU, Geneva, Switzerland.

ITU (1986), Radio Regulations (Edition of 1982, Revised in 1985 and 1986), ITU, Geneva, Switzerland (Appendix 30, in particular).

APPENDIX A. THE MNEBSA SUBROUTINE PACKAGE

The MNEBSA subroutine package determines the minimum elliptical beam of a satellite antenna for a satellite longitude position and a service area. It determines the five parameters of the minimum elliptical beam used in Appendix 30 to the ITU (International Telecommunication Union) Radio Regulations to specify an elliptical beam; i.e., the longitude and latitude of the aim point on the surface of the Earth, the major-axis and minor-axis beamwidths, and the orientation angle of the major axis relative to a line parallel to the equatorial plane in the plane of wave front.

The basic algorithm of the MNEBSA subroutine package presented in this appendix is essentially the same as the algorithm of its original version developed by Akima, in "A method for determining the minimum elliptical beam of a satellite antenna," NTIA Report 81-88 (NTIS Order No. PB82-153966). The basic algorithm of the package is an exhaustive and binary search for the minimum elliptical beam with respect to the longitude and latitude of the aim point, the orientation angle of the major axis of the ellipse, and the axial ratio of the ellipse (i.e., the ratio of the minor-axis radius to the major-axis radius).

This package consists of six subroutines, i.e., the MNEBSA, MNEBTL, MNEBAN, MNEBAR, MNEBEP, and MNEBOA subroutines. The MNEBSA and MNEBTL subroutines interface the user. The MNEBSA subroutine is the master subroutine of the package and determines the beam parameters of the minimum elliptical beam for a satellite longitude position and a service area. The MNEBTL subroutine calculates the antenna gain tolerance for a given elliptical beam and a service area. All other subroutines are the supporting subroutines, called by the MNEBSA or MNEBTL subroutine or by both.

In addition to the minimum elevation angle of the satellite to be mentioned in the following paragraph, the user of this subroutine package can specify the minimum beamwidth, pointing error, and orientation-angle error.

Modifications of the latest version of the MNEBSA subroutine package from its original version include

(1) conversion from 1966 Fortran to 1977 Fortran;

(2) inclusion, as an input argument of the MNEBSA subroutine, of minimum elevation angle of the satellite for which the minimum elliptical beam is to be determined;

(3) inclusion, as an output argument of the MNEBSA subroutine, of elevation angle of the satellite for the service area;

(4) inclusion, as an output argument of the MNEBTL subroutine, of the antenna gain tolerance in decibels;

(5) elimination of the possibility of obtaining an invisible aim point for a very low elevation angle of satellite by skipping the calculations in the MNEBAN and MNEBAR subroutines for an invisible candidate aim point; and

(6) improvement in calculation for a very small or narrow service area, achieved by introducing the concept of absolute minimum beamwidth and by minimizing the following two products, i.e.,

   (a) the product of the greater of the major-axis beamwidth and the absolute minimum beamwidth times the greater of the minor-axis beamwidth and the absolute minimum beamwidth, and

   (b) the product of the greater of the major-axis beamwidth and the user-specified minimum beamwidth times the greater of the minor-axis beamwidth and the user-specified minimum beamwidth.

There are also some minor improvements in readability of the program codes and computational efficiency.

This subroutine package is written in ANSI (American National Standards Institute) Standard Fortran (Publication X3.9-1978, ANSI, 345 East 47th Street, New York, NY 10017).

A Fortran listing of the subroutine package follows. The user information of each subroutine, including the description of the input and output arguments, is given in the beginning of each subroutine.

```
      SUBROUTINE  MNEBSA(EAMN,BWMN,PTER,OAER,
     1                   SLON,NE,ELON,ELAT,EELV,
     2                   ELVA,ALON,ALAT,BWMA,BWMI,ORAN,AREA)
C
C This subroutine determines the minimum elliptical beam of a
C geostationary satellite antenna, given the orbital position of
C the satellite and the locations of the earth points (also
C called the polygon points) for the service area.
C
C U.S. Department of Commerce, NTIA/ITS.
C Version of 87/12/31.
C
C The maximum number of earth points in a service area that can
C be processed by this subroutine is limited by the NEMX constant
C defined in the first parameter statement.  The user can change
C this number at will.
C
C The Input Arguments are
C   EAMN   - minimum value of the elevation angle of the
C            satellite (in degrees), for which this subroutine
C            is to determine the minimum elliptical beam,
C   BWMN   - minimum beamwidth (in degrees),
C   PTER   - pointing error (in degrees),
C   OAER   - orientation-angle error (in degrees),
C   SLON   - longitude of the satellite (in degrees),
C   NE     - number of earth points,
C   ELON   - array of dimension NE that contains the longi-
C            tudes of the earth points (in degrees),
C   ELAT   - array of dimension NE that contains the lati-
C            tudes of the earth points (in degrees),
C   EELV   - array of dimension NE that contains the eleva-
C            tions of the earth points (in meters).
C When a negative value is given to PTER or OAER, its absolute
C value will be used as PTER or OAER.
C
C The Output Arguments are
C   ELVA   - elevation angle of the satellite for the service
C            area (in degrees),
C   ALON   - longitude of the aim point (or boresight) of the
C            elliptical beam of the satellite antenna on the
C            surface of the Earth (in degrees),
C   ALAT   - latitude of the aim point (or boresight) of the
C            elliptical beam of the satellite antenna on the
C            surface of the Earth (in degrees),
C   BWMA   - major-axis beamwidth of the elliptical beam
C            (in degrees),
C   BWMI   - minor-axis beamwidth of the elliptical beam
C            (in degrees),
C   ORAN   - orientation angle of the ellipse (in degrees),
C            i.e., in a plane normal to the beam axis, the
C            angle measured counterclockwise from a line
C            parallel to the equatorial plane to the major
C            axis of the ellipse to the nearest degree,
C   AREA   - area (or solid angle) of the beam (in degrees
```

```
C            squared).
C When the calculated ELVA value is less than the given EAMN
C value, all output arguments except ELVA will be set to zero.
C
C This subroutine calls the MNEBAN and MNEBAR subroutines.
C
C Specification Statements
      PARAMETER   (NEMX=25)
      DIMENSION   ELON(*),ELAT(*),EELV(*)
      DIMENSION   XE(NEMX),YE(NEMX),ZE(NEMX),
     1            ALFA(NEMX),BETA(NEMX),
     2            XDPSQN(NEMX),YDPSQN(NEMX),
     3            XDPSQR(3,NEMX),YDP(3,NEMX)
      PARAMETER   (RE0=6370.997,RGO=42157.197,ZS=0.0)
      PARAMETER   (JLIN1=64,NR1=3,JLIN2=32,NR2=5)
      DIMENSION   JLNP(30),JLTP(30)
      SAVE  INIT,PI,CFDTR,CFRTD,RNGSQM,AJLIN1
      DATA  INIT/0/
C Calculation
C Universal Constants
   10 IF (INIT.LE.0)  THEN
        INIT=1
        PI=2.0*ATAN2(1.0,0.0)
        CFDTR=PI/180.0
        CFRTD=180.0/PI
        RNGSQM=RGO*RGO-RE0*RE0
        AJLIN1=REAL(JLIN1)
      END IF
C Earth-center coordinates of the satellite
   20 XS=RGO*COS(SLON*CFDTR)
      YS=RGO*SIN(SLON*CFDTR)
C Earth-center coordinates of the earth points
      DO 21  JE=1,NE
        RE=RE0+EELV(JE)/1000.0
        COSELA=COS(ELAT(JE)*CFDTR)
        XE(JE)=RE*COSELA*COS(ELON(JE)*CFDTR)
        YE(JE)=RE*COSELA*SIN(ELON(JE)*CFDTR)
        ZE(JE)=RE*SIN(ELAT(JE)*CFDTR)
   21 CONTINUE
C Elevation angle of the satellite for the service area
   30 DO 31  JE=1,NE
        XSE=XS-XE(JE)
        YSE=YS-YE(JE)
        ZSE=  -ZE(JE)
        SE=SQRT(XSE*XSE+YSE*YSE+ZSE*ZSE)
        SP=XSE*XE(JE)+YSE*YE(JE)+ZSE*ZE(JE)
        SNEA=SP/(SE*RE)
        IF (JE.EQ.1)  THEN
          SNEAI=SNEA
        ELSE
          SNEAI=MIN(SNEA,SNEAI)
        END IF
   31 CONTINUE
      SNEAI=MAX(-1.0,MIN(SNEAI,1.0))
```

```
      ELVA=ASIN(SNEAI)*CFRTD
      IF (ELVA.LT.EAMN)        GO TO 800
C Range of aim point
   50 SLONI=SLON
      DO 51  JE=1,NE
        ELONJ=ELON(JE)
        IF (ELONJ.GT.SLONI+180.0)   ELONJ=ELONJ-360.0
        IF (ELONJ.LT.SLONI-180.0)   ELONJ=ELONJ+360.0
        ELATJ=ELAT(JE)
        IF (JE.EQ.1)  THEN
          ALONMN=ELONJ
          ALONMX=ELONJ
          ALATMN=ELATJ
          ALATMX=ELATJ
        ELSE
          ALONMN=MIN(ELONJ,ALONMN)
          ALONMX=MAX(ELONJ,ALONMX)
          ALATMN=MIN(ELATJ,ALATMN)
          ALATMX=MAX(ELATJ,ALATMX)
        END IF
   51 CONTINUE
C Search for minimum elliptical beam without pointing error,
C pointing error, orientation-angle round off, and orientation-
C angle error
  100 PDT1MI=1.0E+6
      PDT2MI=1.0E+6
      DO 149  IR=1,NR1
        IF (IR.EQ.1)  THEN
          JLNCT=NINT((ALONMN+ALONMX)/2.0*10.0)+3600
          N=MAX(2,NINT((ALONMX-ALONMN)/2.0*10.0/AJLIN1)+1)
          JLNMN=JLNCT-N*JLIN1
          JLNMX=JLNCT+N*JLIN1
          JLNIN=JLIN1
          JLTCT=NINT((ALATMN+ALATMX)/2.0*10.0)+1800
          N=MAX(2,NINT((ALATMX-ALATMN)/2.0*10.0/AJLIN1)+1)
          JLTMN=JLTCT-N*JLIN1
          JLTMX=JLTCT+N*JLIN1
          JLTIN=JLIN1
        ELSE
          JLNMN=JLNMI-JLNIN
          JLNMX=JLNMI+JLNIN
          JLNIN=JLNIN/2
          JLTMN=JLTMI-JLTIN
          JLTMX=JLTMI+JLTIN
          JLTIN=JLTIN/2
        END IF
        DO 129  JLN=JLNMN,JLNMX,JLNIN
          IF (IR.EQ.1)  THEN
            JLNFLG=0
          ELSE
            JLNFLG=1
            DO 101  JJLN=1,NJLN
              IF (JLN.EQ.JLNP(JJLN))     GO TO 110
  101       CONTINUE
```

```
            JLNFLG=0
          END IF
  110     ALONJ=REAL(JLN-3600)/10.0
          DO 112  JLT=JLTMN,JLTMX,JLTIN
            IF (JLNFLG.NE.0)  THEN
              DO 111  JJLT=1,NJLT
                IF (JLT.EQ.JLTP(JJLT))   GO TO 112
  111         CONTINUE
            END IF
            ALATJ=REAL(JLT-1800)/10.0
            CALL MNEBAN(BWMN,
     1                  XS,YS,ZS,NE,XE,YE,ZE,
     2                  ALONJ,ALATJ,
     3                  OA,AR,PDT1,PDT2,
     4                  ALFA,BETA,XDPSQN,YDPSQN)
            IF ((PDT2.LT.PDT2MI).OR.
     1          (PDT1.LT.PDT1MI.AND.PDT2.LE.PDT2MI))  THEN
              JLNMI=JLN
              JLTMI=JLT
              OAMI=OA
              ARMI=AR
              PDT1MI=PDT1
              PDT2MI=PDT2
            END IF
  112     CONTINUE
  129   CONTINUE
        IF (IR.NE.NR1)  THEN
          NJLN=0
          DO 131  JLN=JLNMN,JLNMX,JLNIN
            NJLN=NJLN+1
            JLNP(NJLN)=JLN
  131     CONTINUE
          NJLT=0
          DO 132  JLT=JLTMN,JLTMX,JLTIN
            NJLT=NJLT+1
            JLTP(NJLT)=JLT
  132     CONTINUE
        END IF
  149 CONTINUE
      ALON0=REAL(MOD(JLNMI+5399,3600)-1799)/10.0
      ALAT0=REAL(MOD(JLTMI+2699,1800)- 899)/10.0
      ORAN0=OAMI
      AR0=ARMI
C Further search for minimum elliptical beam with the pointing
C error, orientation-angle round off, and orientation-angle error
  200 PDT1MI=1.0E+6
      PDT2MI=1.0E+6
      JLNMI=NINT(ALON0*10.0)+3600
      JLNIN=JLIN2
      JLTMI=NINT(ALAT0*10.0)+1800
      JLTIN=JLIN2
      DO 249  IR=1,NR2
        JLNMN=JLNMI-JLNIN
        JLNMX=JLNMI+JLNIN
```

```
      JLNIN=JLNIN/2
      JLTMN=JLTMI-JLTIN
      JLTMX=JLTMI+JLTIN
      JLTIN=JLTIN/2
      DO 229  JLN=JLNMN,JLNMX,JLNIN
        IF (IR.EQ.1)  THEN
          JLNFLG=0
        ELSE
          JLNFLG=1
          DO 201  JJLN=1,NJLN
            IF (JLN.EQ.JLNP(JJLN))     GO TO 210
201       CONTINUE
          JLNFLG=0
        END IF
210     ALONJ=REAL(JLN-3600)/10.0
        DO 212  JLT=JLTMN,JLTMX,JLTIN
          IF (JLNFLG.NE.0)  THEN
            DO 211  JJLT=1,NJLT
              IF (JLT.EQ.JLTP(JJLT))   GO TO 212
211         CONTINUE
          END IF
          ALATJ=REAL(JLT-1800)/10.0
          CALL MNEBAR(BWMN,ABS(PTER),ABS(OAER),
     1                XS,YS,ZS,NE,XE,YE,ZE,
     2                ALONJ,ALATJ,ORAN0,AR0,
     3                A,B,OA,PDT1,PDT2,
     4                ALFA,BETA,XDPSQR,YDP)
          IF ((PDT2.LT.PDT2MI).OR.
     1        (PDT1.LT.PDT1MI.AND.PDT2.LE.PDT2MI))  THEN
            JLNMI=JLN
            JLTMI=JLT
            AMI=A
            BMI=B
            OAMI=OA
            PDT1MI=PDT1
            PDT2MI=PDT2
          END IF
212     CONTINUE
229   CONTINUE
      IF (IR.NE.NR2)  THEN
        NJLN=0
        DO 231  JLN=JLNMN,JLNMX,JLNIN
          NJLN=NJLN+1
          JLNP(NJLN)=JLN
231     CONTINUE
        NJLT=0
        DO 232  JLT=JLTMN,JLTMX,JLTIN
          NJLT=NJLT+1
          JLTP(NJLT)=JLT
232     CONTINUE
      END IF
249 CONTINUE
    ALON=REAL(MOD(JLNMI+5399,3600)-1799)/10.0
    ALAT=REAL(MOD(JLTMI+2699,1800)- 899)/10.0
```

```
      BWMA=2.0*AMI*CFRTD
      BWMI=2.0*BMI*CFRTD
      ORAN=OAMI
C Rounding up of the beamwidths to one-hundredth of a degree
  300 IF (BWMA.LT.BWMN+0.001)  THEN
        BWMA=BWMN
      ELSE
        BWMA=ANINT(BWMA*100.0+0.5)/100.0
      END IF
      IF (BWMI.LT.BWMN+0.001)  THEN
        BWMI=BWMN
      ELSE
        BWMI=ANINT(BWMI*100.0+0.5)/100.0
      END IF
      IF (BWMA.EQ.BWMI)        ORAN=0.0
      AREA=BWMA*BWMI*PI/4.0
      RETURN
C Resetting of the output arguments when the elevation angle of
C the satellite for the service area is below the specified
C minimum value.
  800 ALON=0.0
      ALAT=0.0
      BWMA=0.0
      BWMI=0.0
      ORAN=0.0
      AREA=0.0
      RETURN
      END



      SUBROUTINE  MNEBTL(PTER,OAER,SLON,NE,ELON,ELAT,EELV,
     1                   ALON,ALAT,BWMA,BWMI,ORAN,
     2                   ELVA,EOFA,EORA,TLAN,TLGA)
C
C This subroutine calculates the tolerances, and other related
C variables, of earth points (also called the polygon points)
C for the service area, relative to an elliptical beam of a
C geostationary satellite antenna.
C
C U.S. Department of Commerce, NTIA/ITS.
C Version of 87/12/31.
C
C The maximum number of earth points in a service area that can
C be processed by this subroutine is limited by the NEMA constant
C defined in the first parameter statemnet.  The user can change
C this number at will.
C
C The Input Arguments are
C   PTER   - pointing error (in degrees),
C   OAER   - orientation-angle error (in degrees),
C   SLON   - longitude of the satellite (in degrees),
C   NE     - number of earth points,
C   ELON   - array of dimension NE that contains the longi-
```

```
C             tudes of the earth points (in degrees),
C    ELAT   - array of dimension NE that contains the lati-
C             tudes of the earth points (in degrees),
C    EELV   - array of dimension NE that contains the eleva-
C             tions of the earth points (in meters),
C    ALON   - longitude of the aim point on the surface of
C             the Earth (in degrees),
C    ALAT   - latitude  of the aim point on the surface of
C             the Earth (in degrees),
C    BWMA   - major-axis beamwidth (in degrees),
C    BWMI   - minor-axis beamwidth (in degrees),
C    ORAN   - orientation angle of the ellipse (in degrees),
C             i.e., in a plane normal to the beam axis, the
C             angle measured counterclockwise from a line
C             parallel to the equatorial plane to the major
C             axis of the ellipse.
C
C The following Output Arguments are all arrays of dimention NE
C where the calculated values of the following variables are to
C be stored.
C    ELVA   - elevation angle of the satellite seen from the
C             earth point (in degrees),
C    EOFA   - off-axis angle of the earth point (in degrees),
C    EORA   - orientation angle of the earth point (in
C             degrees), i.e., the angle measured counter-
C             clockwise, in a plane normal to the beam axis,
C             from a line parallel to the equatorial plane to
C             the projection of the vector from the aim point
C             to the earth point,
C    TLAN   - tolerance of the earth point in terms of angel
C             (in degrees), i.e., the distance between the
C             earth point and the circumference of the ellipse
C             expressed in the angle seen from the satellite
C             antenna,
C    TLGA   - tolerance of the earth point in terms of antenna
C             gain (in decibels), i.e., the difference in
C             antenna gain between the earth point and the
C             point on the circumference of the ellipse,
C             assuming the parabolic roll-off curve.
C
C This subroutine calls the MNEBEP and MNEBOA subroutines.
C
C Specification Statements
      PARAMETER   (NEMX=25)
      DIMENSION   ELON(*),ELAT(*),EELV(*)
      DIMENSION   ELVA(*),EOFA(*),EORA(*),TLAN(*),TLGA(*)
      DIMENSION   XE(NEMX),YE(NEMX),ZE(NEMX),
     1            ALFA(NEMX),BETA(NEMX)
      PARAMETER   (NPED=72)
      DIMENSION   ALFAPE(NPED),BETAPE(NPED),
     1            XAPE(NPED),YAPE(NPED),ZAPE(NPED)
      PARAMETER   (REO=6370.997,RGO=42157.197,ZS=0.0)
      SAVE  INIT,PI,CFDTR,CFRTD,RNGSQM
      DATA  INIT/0/
```

```
C Calculation
C Universal constants
   10 IF (INIT.LE.0)  THEN
         INIT=1
         PI=2.0*ATAN2(1.0,0.0)
         CFDTR=PI/180.0
         CFRTD=180.0/PI
         RNGSQM=RGO*RGO-RE0*RE0
      END IF
C Earth-center coordinates of the satellite
   20 XS=RGO*COS(SLON*CFDTR)
      YS=RGO*SIN(SLON*CFDTR)
C Earth-center coordinates of the aim point
   25 COSALA=COS(ALAT*CFDTR)
      XA=RE0*COSALA*COS(ALON*CFDTR)
      YA=RE0*COSALA*SIN(ALON*CFDTR)
      ZA=RE0*SIN(ALAT*CFDTR)
C Earth-center coordinates of the earth points
   30 DO 31  JE=1,NE
         RE=RE0+EELV(JE)/1000.0
         COSELA=COS(ELAT(JE)*CFDTR)
         XE(JE)=RE*COSELA*COS(ELON(JE)*CFDTR)
         YE(JE)=RE*COSELA*SIN(ELON(JE)*CFDTR)
         ZE(JE)=RE*SIN(ELAT(JE)*CFDTR)
   31 CONTINUE
C Elevation angles of the satellite seen from the earth points
   40 DO 41  JE=1,NE
         XSE=XS-XE(JE)
         YSE=YS-YE(JE)
         ZSE=  -ZE(JE)
         SE=SQRT(XSE*XSE+YSE*YSE+ZSE*ZSE)
         SP=XSE*XE(JE)+YSE*YE(JE)+ZSE*ZE(JE)
         SINEA=MAX(-1.0,MIN(SP/(SE*RE),1.0))
         ELVA(JE)=ASIN(SINEA)*CFRTD
   41 CONTINUE
C Off-axis angles and orientation angles of the earth points
   50 CALL MNEBOA(XS,YS,ZS,XA,YA,ZA,NE,XE,YE,ZE, ALFA,BETA)
      DO 51  JE=1,NE
        EOFA(JE)=ALFA(JE)*CFRTD
        EORA(JE)=BETA(JE)*CFRTD
   51 CONTINUE
C Semi-axis beamwidths
   60 A=BWMA*CFDTR/2.0
      B=BWMI*CFDTR/2.0
      ASQ=A*A
      BSQ=B*B
C Tolerances of the earth points in terms of angle
   70 DO 79  JE=1,NE
         DMN=1.0E+6
         DO 72  JOE=1,3
           OAJ=(ORAN+OAER*REAL(JOE-2))*CFDTR
           XEDP=ABS(ALFA(JE)*COS(BETA(JE)-OAJ))
           YEDP=ABS(ALFA(JE)*SIN(BETA(JE)-OAJ))
           X1DP=MIN(XEDP,A)
```

```
          X2DP=A/B*SQRT(MAX(0.0,BSQ-YEDP**2))
          DO 71  JXDP=1,21
            XDP=X1DP+(X2DP-X1DP)*REAL(JXDP-1)/100.0
            YDP=B/A*SQRT(MAX(0.0,ASQ-XDP**2))
            DMN=MIN(SQRT((XDP-XEDP)**2+(YDP-YEDP)**2),DMN)
   71     CONTINUE
   72   CONTINUE
        IF (((XEDP/A)**2+(YEDP/B)**2).LE.1.0)  THEN
          TLAN(JE)= DMN*CFRTD-ABS(PTER)
        ELSE
          TLAN(JE)=-DMN*CFRTD-ABS(PTER)
        END IF
   79 CONTINUE
C Tolerances of the earth points in terms of antenna gain
   80 DO 81  JPED=1,NPED
        ALFAPE(JPED)=PTER*CFDTR
        BETAPE(JPED)=2.0*PI*REAL(JPED)/REAL(NPED)
   81 CONTINUE
      CALL MNEBEP(XS,YS,ZS,XA,YA,ZA,NPED,ALFAPE,BETAPE,
     1            XAPE,YAPE,ZAPE)
       DO 89  JPED=1,NPED
         CALL MNEBOA(XS,YS,ZS,XAPE(JPED),YAPE(JPED),ZAPE(JPED),
     1               NE,XE,YE,ZE, ALFA,BETA)
         DO 87  JOE=1,3
           OAJ=(ORAN+OAER*REAL(JOE-2))*CFDTR
           DO 86  JE=1,NE
             IF (JPED.EQ.1.AND.JOE.EQ.1)   TLGA(JE)=0.0
             COSB=COS(BETA(JE)-OAJ)
             COSBSQ=COSB*COSB
             SINBSQ=1.0-COSBSQ
             RSQ=(ALFA(JE)**2)*(COSBSQ/ASQ+SINBSQ/BSQ)
             TLGA(JE)=MAX(TLGA(JE),RSQ)
             IF (JPED.EQ.NPED.AND.JOE.EQ.3)
     1         TLGA(JE)=3.0*(1.0-TLGA(JE))
    86     CONTINUE
    87   CONTINUE
    89 CONTINUE
       RETURN
       END



       SUBROUTINE  MNEBAN(BWMN,
      1                   XS,YS,ZS,NE,XE,YE,ZE,
      2                   ALON,ALAT,
      3                   OAM,ARM,PDT1M,PDT2M,
      4                   ALFA BETA,XDPSQ,YDPSQ)
C
C This subroutine calculates, for an aim point, the minimum
C value (minimized over the orientation angle and axial ratio
C values) of the product of the semimajor and semiminor axes of
C the elliptical beam of a satellite antenna.  Calculation does
C not include the pointing error, orientation-angle round off,
C or orientation-angle error.
```

```
C
C U.S. Department of Commerce, NTIA/ITS.
C Version of 87/12/31.
C
C The Input Arguments are
C   BWMN   - minimum beamwidth (in degrees),
C   XS, YS, ZS
C          - earth-center coordinates of the satellite,
C   NE     - number of earth points,
C   XE, YE, ZE
C          - arrays of dimension NE containing the earth-
C            center coordinates of the earth points,
C   ALON   - longitude of the aim point on the surface of
C            the earth (in degrees),
C   ALAT   - latitude  of the aim point on the surface of
C            the earth (in degrees).
C
C The Output Arguments are
C   OAM    - optimum value of the orientation angle of the
C            ellipse (in degrees),
C   ARM    - optimum value of the axial ratio,
C   PDT1M  - minimum value of the product of the semimajor
C            axis times semiminor axis of the elliptical
C            beam without minimum beamwidth restriction
C            (in radians squared),
C   PDT2M  - minimum value of the product of the semimajor
C            axis times semiminor axis of the elliptical
C            beam with minimum beamwidth restriction
C            (in radians squared).
C
C Other Arguments are
C   ALFA, BETA, XDPSQ, YDPSQ
C          - arrays of dimension NE used internally as work
C            areas.
C
C This subroutine calls the MNEBOA subroutine.
C
C Specification Statements
      DIMENSION   XE(*),YE(*),ZE(*)
      DIMENSION   ALFA(*),BETA(*),XDPSQ(*),YDPSQ(*)
      PARAMETER   (RE0=6370.997,RGO=42157.197)
      PARAMETER   (BWMN0=0.0999)
      PARAMETER   (NOA=1024,JOAIN0=128,NRO=4)
      PARAMETER   (NAR=1024,JARIN0=128,NRA=6)
      SAVE  INIT,PI,CFDTR,RNGSQM,SBWMN0,ANOA,ANAR
      DATA  INIT/0/
C Calculation
C Universal constants
   10 IF (INIT.LE.0)  THEN
         INIT=1
         PI=2.0*ATAN2(1.0,0.0)
         CFDTR=PI/180.0
         RNGSQM=RGO*RGO-RE0*RE0
         SBWMN0=BWMN0*CFDTR/2.0
```

```
        ANOA=REAL(NOA)
        ANAR=REAL(NAR)
      END IF
C Minimum semi-beamwidth in radians
      SBWMN=MAX(SBWMN0,BWMN*CFDTR/2.0)
C Earth-center coordinates of aim point
   20 COSALA=COS(ALAT*CFDTR)
      XA=RE0*COSALA*COS(ALON*CFDTR)
      YA=RE0*COSALA*SIN(ALON*CFDTR)
      ZA=RE0*SIN(ALAT*CFDTR)
      RNGSQ=(XS-XA)**2+(YS-YA)**2+(ZS-ZA)**2
      IF (RNGSQ.GT.RNGSQM)     GO TO 80
C Off-axis and orientation angles of the earth points
   30 CALL MNEBOA(XS,YS,ZS,XA,YA,ZA,NE,XE,YE,ZE,ALFA,BETA)
C Main DO-loops (2) with respect to the orientation angle
   40 PDT1MI=1.0E+6
      PDT2MI=1.0E+6
      DO 69  IRO=1,NRO
        IF (IRO.EQ.1)  THEN
          JOAMN=0
          JOAMX=NOA
          JOAIN=JOAIN0
        ELSE
          IF (IRO.NE.2)        JOAIN=JOAIN/2
          JOAMN=JOAMI-JOAIN/2
          JOAMX=JOAMI+JOAIN/2
        END IF
        DO 68  JOA=JOAMN,JOAMX,JOAIN
          OAJ=REAL(MOD(JOA+NOA,NOA))/ANOA*PI
          DO 41  JE=1,NE
            XDPSQ(JE)=(ALFA(JE)*COS(BETA(JE)-OAJ))**2
            YDPSQ(JE)=ALFA(JE)**2-XDPSQ(JE)
   41     CONTINUE
C Inner DO-loops (2) with respect to the axial ratio
   50     DO 53  IRA=1,NRA
            IF (IRA.EQ.1)  THEN
              JARMN=0
              JARMX=NAR
              JARIN=JARIN0
            ELSE
              IF (IRA.NE.2)    JARIN=JARIN/2
              JARMN=JARMI-JARIN/2
              JARMX=JARMI+JARIN/2
            END IF
            IF (JARIN.NE.0)  THEN
              DO 52  JAR=JARMN,JARMX,JARIN
                JARP=MIN(MAX(1,JAR),NAR)
                RAR=ANAR/REAL(JARP)
                RARSQ=RAR*RAR
                RSQMX=0.0
                DO 51  JE=1,NE
                  RSQMX=MAX(RSQMX,XDPSQ(JE)+YDPSQ(JE)*RARSQ)
   51           CONTINUE
                A=SQRT(RSQMX)
```

```
                B=A/RAR
                A=MAX(SBWMN0,A)
                B=MAX(SBWMN0,B)
                PDT1=A*B
                A=MAX(SBWMN,A)
                B=MAX(SBWMN,B)
                PDT2=A*B
                IF ((PDT2.LT.PDT2MI).OR.
     1              (PDT1.LT.PDT1MI.AND.PDT2.LE.PDT2MI))  THEN
                  JOAMI=JOA
                  JARMI=JARP
                  PDT1MI=PDT1
                  PDT2MI=PDT2
                END IF
   52         CONTINUE
            END IF
   53     CONTINUE
   68   CONTINUE
   69 CONTINUE
      OAM=REAL(MOD(JOAMI,NOA))/ANOA*180.0
      ARM=REAL(JARMI)/ANAR
      PDT1M=PDT1MI
      PDT2M=PDT2MI
      RETURN
C Case of an invisible aim point
   80 PDT1M=1.0E+6
      PDT2M=1.0E+6
      RETURN
      END



      SUBROUTINE  MNEBAR(BWMN,PTER,OAER,
     1                   XS,YS,ZS,NE,XE,YE,ZE,
     2                   ALON,ALAT,ORAN0,AR0,
     3                   AM,BM,OAM,PDT1M,PDT2M,
     4                   ALFA,BETA,XDPSQ,YDP)
C
C This subroutine calculates, for an aim point, the minimum
C value (minimized over the orientation angle and axial ratio
C values) of the product of the semimajor and semiminor axes of
C the elliptical beam of a satellite antenna.  Calculation does
C include the pointing error, orientation-angle round off, and
C orientation-angle error.
C
C U.S. Department of Commerce, NTIA/ITS.
C Version of 87/12/31.
C
C The Input Arguments are
C   BWMN   - minimum beamwidth (in degrees),
C   PTER   - pointing error (in degrees),
C   OAER   - orientation-angle error (in degrees),
C   XS, YS, ZS
C          - earth-center coordinates of the satellite,
```

```
C   NE      - number of earth points,
C   XE, YE, ZE
C           - arrays of dimension NE containing the earth-
C             center coordinates of the earth points,
C   ALON    - longitude of aim point on the surface of
C             the Earth (in degrees),
C   ALAT    - latitude  of aim point on the surface of
C             the Earth (in degrees),
C   ORANO   - optimum value of the orientation angle of the
C             ellipse without pointing error, orientation-
C             angle round off, and orientation-angle error
C             (in degrees),
C   ARO     - optimum value of the axial ratio of the ellipse
C             without pointing error, orientation-angle round
C             off, and orientation-angle error.
C
C The Output Arguments are
C   AM      - optimum value of the semimajor beamwidth
C             (in radians),
C   BM      - optimum value of the semiminor beamwidth
C             (in radians),
C   OAM     - optimum value of the orientation angle of the
C             ellipse (in degrees),
C   PDT1M   - minimum value of the product of the semimajor
C             axis times semiminor axis of the elliptical
C             beam without minimum beamwidth restriction
C             (in radians squared),
C   PDT2M   - minimum value of the product of the semimajor
C             axis times semiminor axis of the elliptical
C             beam with minimum beamwidth restriction
C             (in radians squared).
C
C Other Arguments are
C   ALFA, BETA
C           - arrays of dimension NE used internally as work
C             areas,
C   XDPSQ, YDP
C           - doubly dimensioned arrays of dimension (3,NE)
C             used internally as work areas.
C
C This subroutine calls the MNEBOA subroutine.
C
C Specification Statements
      DIMENSION   XE(*),YE(*),ZE(*)
      DIMENSION   ALFA(*),BETA(*),XDPSQ(3,*),YDP(3,*)
      PARAMETER   (REO=6370.997,RGO=42157.197)
      PARAMETER   (BWMNO=0.0999)
      PARAMETER   (JOAINO=32,NRO=6)
      PARAMETER   (NAR=1024,JARINO=128,NRA=8)
      SAVE  INIT,CFDTR,RNGSQM,SBWMNO,ANAR
      DATA  INIT/0/
C Calculation
C Universal constants
   10 IF (INIT.LE.0)  THEN
```

```
        INIT=1
        CFDTR=ATAN2(1.0,0.0)/90.0
        RNGSQM=RGO*RGO-RE0*RE0
        SBWMN0=BWMN0*CFDTR/2.0
        ANAR=REAL(NAR)
      END IF
C Minimum semi-beamwidth in radians
      SBWMN=MAX(SBWMN0,BWMN*CFDTR/2.0)
C Pointing error in radians
      APTER=ABS(PTER)*CFDTR
C Earth-center coordinates of aim point
   20 COSALA=COS(ALAT*CFDTR)
      XA=RE0*COSALA*COS(ALON*CFDTR)
      YA=RE0*COSALA*SIN(ALON*CFDTR)
      ZA=RE0*SIN(ALAT*CFDTR)
      RNGSQ=(XS-XA)**2+(YS-YA)**2+(ZS-ZA)**2
      IF (RNGSQ.GT.RNGSQM)     GO TO 80
C Off-axis and orientation angles of the earth points
   30 CALL MNEBOA(XS,YS,ZS,XA,YA,ZA,NE,XE,YE,ZE,ALFA,BETA)
C Main DO-loop with respect to the orientation angle
   40 PDT1MI=1.0E+6
      PDT2MI=1.0E+6
      DO 69  IRO=1,NRO
        IF (IRO.EQ.1)  THEN
          JOA0=NINT(ORAN0)
          JOAMN=JOA0-JOAIN0
          JOAMX=JOA0+JOAIN0
          JOAIN=JOAIN0
        ELSE
          IF (IRO.NE.2)   JOAIN=JOAIN/2
          JOAMN=JOAMI-JOAIN/2
          JOAMX=JOAMI+JOAIN/2
        END IF
        DO 68  JOA=JOAMN,JOAMX,JOAIN
          DO 42  JOE=1,3
            OAJ=(REAL(JOA)+ABS(OAER)*REAL(JOE-2))*CFDTR
            DO 41  JE=1,NE
              YDP0=ABS(ALFA(JE)*SIN(BETA(JE)-OAJ))
              XDPSQ(JOE,JE)=ALFA(JE)**2-YDP0**2
              YDP(JOE,JE)=YDP0+APTER
   41       CONTINUE
   42     CONTINUE
C Inner DO-loops (2) with respect to the axial ratio
   50     DO 54  IRA=1,NRA
            IF (IRA.EQ.1)  THEN
              JAR0=NINT(AR0*ANAR)
              JARMN=JAR0-JARIN0
              JARMX=JAR0+JARIN0
              JARIN=JARIN0
            ELSE
              IF (IRA.NE.2)      JARIN=JARIN/2
              JARMN=JARMI-JARIN/2
              JARMX=JARMI+JARIN/2
            END IF
```

```
            IF (JARIN.NE.0)  THEN
              DO 53  JAR=JARMN,JARMX,JARIN
                JARP=MIN(MAX(1,JAR),NAR)
                RAR=ANAR/REAL(JARP)
                RSQMX=0.0
                DO 52  JOE=1,3
                  DO 51  JE=1,NE
                    YDPC=YDP(JOE,JE)*RAR-APTER
                    RSQ=XDPSQ(JOE,JE)+YDPC**2
                    RSQMX=MAX(RSQMX,RSQ)
   51             CONTINUE
   52           CONTINUE
                A=SQRT(RSQMX)+APTER
                B=A/RAR
                A=MAX(SBWMN0,A)
                B=MAX(SBWMN0,B)
                PDT1=A*B
                A=MAX(SBWMN,A)
                B=MAX(SBWMN,B)
                PDT2=A*B
                IF ((PDT2.LT.PDT2MI).OR.
     1              (PDT1.LT.PDT1MI.AND.PDT2.LE.PDT2MI))  THEN
                  AMI=A
                  BMI=B
                  JOAMI=JOA
                  JARMI=JARP
                  PDT1MI=PDT1
                  PDT2MI=PDT2
                END IF
   53         CONTINUE
            END IF
   54     CONTINUE
   68   CONTINUE
   69 CONTINUE
   70 AM=AMI
      BM=BMI
      OAM=MOD(JOAMI+180,180)
      PDT1M=PDT1MI
      PDT2M=PDT2MI
      RETURN
C Case of an invisible aim point
   80 PDT1M=1.0E+6
      PDT2M=1.0E+6
      RETURN
      END



      SUBROUTINE  MNEBEP(XS,YS,ZS,XA,YA,ZA,NE,ALFA,BETA,
     1                   XE,YE,ZE)
C
C This subroutine calculates the earth-center coordinates of the
C earth points, given the off-axis angles and orientation angles
C of the earth points.
```

```
C
C U.S. Department of Commerce, NTIA/ITS.
C Version of 87/12/31.
C
C The Input Arguments are
C   XS, YS, ZS
C           - earth-center coordinates of the satellite,
C   XA, YA, ZA
C           - earth-center coordinates of aim point on
C             the surface of the Earth,
C   NE      - number of earth points,
C   ALFA    - array of dimension NE containing the off-axis
C             angles of the earth points (in radians),
C   BETA    - array of dimension NE containing the orientation
C             angles of the earth points (in radians).
C
C The Output Arguments are
C   XE, YE, ZE
C           - arrays of dimension NE, where the calculated
C             values of the earth-center coordinates of the
C             earth points are to be stored.
C
C Specification Statements
      DIMENSION   ALFA(*),BETA(*)
      DIMENSION   XE(*),YE(*),ZE(*)
C Calculation of the coefficients of coordinate transformation
   10 A31=XS-XA
      A32=YS-YA
      A33=ZS-ZA
      ZSP=SQRT(A31*A31+A32*A32+A33*A33)
      A31=A31/ZSP
      A32=A32/ZSP
      A33=A33/ZSP
      AA=SQRT(A31*A31+A32*A32)
      A11=-A32/AA
      A12=A31/AA
C     A13=0.0
      A21=       -A33*A12
      A22=A33*A11
      A23=A31*A12-A32*A11
C Equatorial-plane coordinates of the earth center
   20 XCP=-A11*XA-A12*YA
      YCP=-A21*XA-A22*YA-A23*ZA
      ZCP=-A31*XA-A32*YA-A33*ZA
C Equatorial-plane coordinates of the earth points, and
C Coordinate transformation to the earth-center coordinates
   30 DO 31  JE=1,NE
        COSALF=COS(ALFA(JE))
        SINALF=SIN(ALFA(JE))
        COSBET=COS(BETA(JE))
        SINBET=SIN(BETA(JE))
        BB=(XCP*COSBET+YCP*SINBET)*SINALF+(ZSP-ZCP)*COSALF
        CC=ZSP*(ZSP-ZCP-ZCP)
        DD=BB*BB-CC
```

```
      IF (DD.GE.0.0)  THEN
        RR=BB-SQRT(DD)
      ELSE
        RR=SQRT(CC)
      END IF
      XEP=RR*SINALF*COSBET
      YEP=RR*SINALF*SINBET
      ZEP=ZSP-RR*COSALF
      XE(JE)=A11*XEP+A21*YEP+A31*ZEP+XA
      YE(JE)=A12*XEP+A22*YEP+A32*ZEP+YA
      ZE(JE)=        A23*YEP+A33*ZEP+ZA
   31 CONTINUE
      RETURN
      END



      SUBROUTINE  MNEBOA(XS,YS,ZS,XA,YA,ZA,NE,XE,YE,ZE,
     1                   ALFA,BETA)
C
C This subroutine calculates the off-axis angles and orientation
C angles of the earth points, given the earth-center coordinates
C of the earth points.
C
C U.S. Department of Commerce, NTIA/ITS.
C Version of 87/12/31.
C
C The Input Arguments are
C   XS, YS, ZS
C           - earth-center coordinates of the satellite,
C   XA, YA, ZA
C           - earth-center coordinates of aim point on
C             the surface of the Earth,
C   NE      - number of earth points,
C   XE, YE, ZE
C           - arrays of dimension NE containing the earth-
C             center coordinates of the earth points.
C
C The Cutput Arguments are
C   ALFA    - array of dimension NE, where the calculated
C             values of the off-axis angles of the earth
C             points are to be stored (in radians),
C   BETA    - array of dimension NE, where the calculated
C             values of the orientation angles of the earth
C             points are to be stored (in radians).
C
C Specification Statements
      DIMENSION   XE(*),YE(*),ZE(*)
      DIMENSION   ALFA(*),BETA(*)
C Calculation of the coefficients of coordinate transformation
   10 A31=XS-XA
      A32=YS-YA
      A33=ZS-ZA
      ZSP=SQRT(A31*A31+A32*A32+A33*A33)
```

```
      A31=A31/ZSP
      A32=A32/ZSP
      A33=A33/ZSP
      AA=SQRT(A31*A31+A32*A32)
      A11=-A32/AA
      A12=A31/AA
C     A13=0.0
      A21=       -A33*A12
      A22=A33*A11
      A23=A31*A12-A32*A11
C Coordinate transformation and calculation of the angles
   20 DO 21  JE=1,NE
        XEP=A11*(XE(JE)-XA)+A12*(YE(JE)-YA)
        YEP=A21*(XE(JE)-XA)+A22*(YE(JE)-YA)+A23*(ZE(JE)-ZA)
        ZEP=A31*(XE(JE)-XA)+A32*(YE(JE)-YA)+A33*(ZE(JE)-ZA)
        ALFA(JE)=ATAN2(SQRT(XEP*XEP+YEP*YEP),(ZSP-ZEP))
        IF (ALFA(JE).EQ.0.0)  THEN
          BETA(JE)=0.0
        ELSE
          BETA(JE)=ATAN2(YEP,XEP)
        END IF
   21 CONTINUE
      RETURN
      END
```

APPENDIX B. THE CDMNEB PROGRAM

The CDMNEB program reads in the polygon-point data of service areas from the DFPLGN data file, determines necessary satellite longitude positions for the service areas, determines minimum elliptical beams for the necessary satellite longitude positions for the service areas, and creates (or appends) the DFMNEB data file that stores the data (or beam parameters) of the determined beams.

To run this program, the user must supply the DFPLGN data file as the input data file. The format and the contents of the DFPLGN data file are described in the user information in the listing of the program given below.

It is assumed that the DFMNEB data file will be used later in such a way that an elliptical beam for an arbitrary satellite longitude position will be determined by linear interpolation of the beam parameters of the minimum elliptical beams for the two satellite longitude positions that bound the satellite longitude position in question. Necessary satellite longitude positions for each service area are the satellite longitude positions that are necessary to keep the antenna gain tolerance of the interpolated beam from the DFMNEB data file above -0.1 dB.

This program calls the MNEBSA subroutine to determine the minimum elliptical beam for a satellite longitude position for a service area.

This program can be run either with or without the DFMNEB data file. It creates the DFMNEB data file when it is run for the first time (without the DFMNEB data file). It appends the existing DFMNEB data file when it is run repeatedly (with the DFMNEB data file). The format and the contents of the DFMNEB data file can be transparent to the user if the file is used with the MNEBIP subroutine that is described in Appendix C.

This program is written in ANSI (American National Standards Institute) Standard Fortran (Publication X3.9-1978, ANSI, 345 East 47th Street, New York, NY 10017).

A Fortran listing of the program follows. The user information of the program, including the description of the input and output data files, is given in the beginning of the program.

```
      PROGRAM  CDMNEB
C
C This program creates or appends the data file of the minimum
C elliptical beams of satellite antennas for given service areas.
C
C U.S. Department of Commerce, NTIA/ITS.
C Version of 87/12/31.
C
C The created data for each service area include data for all
C necessary satellite longitude positions.  The data for each
C service area and for each satellite longitude position include
C the satellite longitude, minimum elevation angle of the
C satellite seen from the service area, and five parameters of
C the elliptical beam (i.e., the longitude and latitude of the
C aim point (or boresight), major-axis and minor-axis beamwidths,
C and orientation angle of the major axis).  To obtain the data
C of elliptical beam for a desired satellite longitude position
C from the data in the created or appended data file, the user is
C expected to interpolate linearly the five parameters of
C elliptical beams for two satellite longitude positions that
C bound or are closest to the desired satellite longitude
C position.
C
C The CDMNEB program opens the following files:
C   DFPLGN - data file containing the polygon data for the
C            service area,
C   DFMNEB - data file to be created or appended to store the
C            calculated minimum elliptical beam parameters
C            (a direct-access, unformatted file with the
C            record length set to 1024 bytes), and
C   CDMOUT - data file where the calculated results are to
C            be written.
C This program does not require the DFMNEB file to start to run.
C When the DFMNEB file does not exist, the program creates it as
C a new file.  When the DFMNEB file exists, the program appends
C the file in such a way that the appended file includes all
C service areas included in the DFPLGN and original DFMNEB files.
C
C This program can be run with a combination of values of the
C following parameters:
C   EAMN - minimum elevation angle of the satellite for the
C          service area (in degrees) for which this program
C          calculates the minimum elliptical beam,
C   BWMN - minimum beamwidth (in degrees),
C   PTER - pointing error (in degrees), and
C   OAER - orientation-angle error (in degrees).
C The user can specify values of these parameters in the first
C line of the DFMNEB file.  If the user does not give the DFMNEB
C file, the program takes the following default values:
C   EAMN -  0.0 (degrees),
C   BWMN -  1.0 (degrees),
C   PTER -  0.1 (degrees), and
C   OAER -  1.0 (degrees).
C
```

```
C The DFPLGN File Format -
C   The first line of the file contains the file name 'DFPLGN'
C   and an optional date in FORMAT (A6,18X,A8).  This line is
C   followed by one or more blocks of data for service areas.
C   The first line of each block of data for a service area
C   contains the following data:
C     (1) NMSA, i.e., the name of the service area,
C     (2) NRPP, i.e., the number of polygon points, and
C     (3) an optional descriptive name of the service area
C   in FORMAT (A8,8X,I3,5X,A40).
C   The first line of each block of data is followed by one or
C   more lines, each of which contains up to ten longitude
C   values of polygon points in FORMAT (10F8.2), followed by one
C   or more lines, each of which contains up to ten latitude
C   values of polygon points in FORMAT (10F8.2).
C   Note that the optional date and descriptive name of the
C   service area in this file will not be used when the program
C   is run.  They are for bookkeeping purposes only.
C
C The DFMNEB File Contents -
C   The first record of the file contains the following data:
C     (1) the file name 'DFMNEB',
C     (2) NRSA, i.e., the total number of service areas
C         included in this file,
C     (3) EAMN, i.e., the minimum elevation angle of the
C         satellite for the service area (in degrees) for
C         which this program calculates the minimum
C         elliptical beam,
C     (4) BWMN, i.e., the minimum beamwidth (in degrees),
C     (5) PTER, i.e., the pointing error (in degrees), and
C     (6) OAER, i.e., the orientation-angle error
C         (in degrees).
C   Each of the following NRSA records contains the following
C   data for each service area:
C     (1) NMSA, i.e., the name of the service area,
C     (2) NRSL,  i.e., number of satellite longitude
C         positions,
C     (3) SLON, i.e., the satellite longitudes (in degrees),
C     (4) ELVA, i.e., the elevation angles of the satellite
C         for the service area (in degrees), defined as the
C         minimum value of the elevation angle of the
C         satellite seen from the earth points of the
C         service area,
C     (5) ALON, i.e., the longitudes of the aim points
C         (or boresight) of the elliptical beams of the
C         satellite antenna on the surface of the Earth
C         (in degrees),
C     (6) ALAT, i.e., the latitudes of the aim points
C         (or boresight) of the elliptical beams of the
C         satellite antenna on the surface of the Earth
C         (in degrees),
C     (7) BWMA, i.e., the major-axis beamwidths of the
C         elliptical beams (in degrees),
C     (8) BWMI, i.e., the minor-axis beamwidths of the
```

```
C         elliptical beams (in degrees), and
C      (9) ORAN, i.e., the orientation angles of the ellipse
C         (in degrees), each defined in a plane normal to
C         the beam axis as the angle measured counterclock-
C         wise from a line parallel to the equatorial plane
C         to the major axis of the ellipse to the nearest
C         degree.
C   The last (NRSA+127)/128 file(s) contain(s) the names of all
C   NRSA service areas.
C
C This program calls the MNEBSA subroutine.
C
C Currently, this program can process up to 50 polygon points
C for each service area.  The user can change the maximum number
C of polygon points at will by changing the value of the NRPPMX
C parameter defined in the first PARAMETER statement, provided
C that it does not exceed the value of the same parameter
C defined in the MNEBSA subroutine, called by this program.
C
C Specification Statements
C - PARAMETER Statements
      PARAMETER   (NRPPMX=50)
      PARAMETER   (NRSAMX=512)
      PARAMETER   (NRSLMX=42)
      PARAMETER   (EAMNDF= 0.0)
      PARAMETER   (BWMNDF= 1.0)
      PARAMETER   (PTERDF= 0.1)
      PARAMETER   (OAERDF= 1.0)
      PARAMETER   (REO=6370.997, RGO=42157.197, ZS=0.0)
C - DIMENSION Statements
      DIMENSION   ELON(NRPPMX),ELAT(NRPPMX),EELV(NRPPMX)
      DIMENSION   XE(NRPPMX),YE(NRPPMX),ZE(NRPPMX)
      DIMENSION   NMSA(NRSAMX)
      DIMENSION   SLON(NRSLMX),ELVA(NRSLMX),
     1            ALON(NRSLMX),ALAT(NRSLMX),
     2            BWMA(NRSLMX),BWMI(NRSLMX),
     3            ORAN(NRSLMX),INSO(NRSLMX)
      DIMENSION   ICALC(540)
C - TYPE Statements
      CHARACTER*8 NMDFMN,NMSA,NMSAP
C - DATA INITIALIZATION Statements
      DATA  EELV/NRPPMX*0.0/, IPAGE/0/
C Calculation
C Universal Constanta
   10 PI=2.0*ATAN2(1.0,0.0)
      CFDTR=PI/180.0
      CFRTD=180.0/PI
C Opens the files.
   20 OPEN  (1,FILE='DFPLGN',STATUS='OLD',ERR=910,IOSTAT=IOS)
      OPEN  (2,FILE='DFMNEB',STATUS='UNKNOWN',
     1       ACCESS='DIRECT',FORM='UNFORMATTED',RECL=1024)
      OPEN  (6,FILE='CDMOUT',FORM='PRINT')
C Reads in (or enters the default values of) the basic parameter
C values.
```

```
   30 READ  (2,REC=1,END=35,ERR=922,IOSTAT=IOS)
     1       NMDFMN,NRSA,EAMN,BWMN,PTER,OAER
      GO TO 39
   35 NMDFMN='DFMNEB  '
      NRSA=0
      EAMN=EAMNDF
      BWMN=BWMNDF
      PTER=PTERDF
      OAER=OAERDF
   39 CONTINUE
C Reads in the names of the service areas included in the DFMNEB
C data file, if any.
   40 IF (NRSA.GT.0)  THEN
        NRRECS=(NRSA+127)/128
        DO 41  IRECS=1,NRRECS
          IREC=1+NRSA+IRECS
          ISAMN=(IRECS-1)*128+1
          ISAMX=MIN(ISAMN+127,NRSA)
          READ  (2,REC=IREC,END=921,ERR=922,IOSTAT=IOS)
     1           (NMSA(ISA),ISA=ISAMN,ISAMX)
   41   CONTINUE
      END IF
C The outermost loop with respect to the service area, NMSAP, in
C the DFPLGN data file  (Statement Nos. 100 - 399)
  100 REWIND 1
      READ  (1,'(1X)',END=911,ERR=912,IOSTAT=IOS)
  110 READ  (1,'(A8,8X,I3)',END=399,ERR=912,IOSTAT=IOS)  NMSAP,NRPP
      IF (NRPP.LE.0)       GO TO 110
      READ  (1,'(10F8.2)',END=911,ERR=912,IOSTAT=IOS)
     1      (ELON(IPP),IPP=1,NRPP)
      READ  (1,'(10F8.2)',END=911,ERR=912,IOSTAT=IOS)
     1      (ELAT(IPP),IPP=1,NRPP)
C Checks if the data for the NMSAP service area is included in
C the DFMNEB file.  If so, skips the calculation.
  120 IF (NRSA.GT.0)  THEN
        DO 121  ISA=1,NRSA
          IF (NMSAP.EQ.NMSA(ISA))   GO TO 110
  121   CONTINUE
      END IF
C New calculation when the data for the NMSAP service area is
C not included in the DFMNEB file  (Statement Nos. 200 - 329)
C Selection of the satellite longitude positions for which the
C beams are to be calculated  (Statement Nos. 200 - 232)
C Calculation of the elevation angles of the satellite and
C setting of the ICALC flags for satellite longitude positions
C for which calculations of the beam parameters are necessary
C in the orbital arc from -179 degrees to 360 degrees
C (Statement Nos. 200 - 213)
  200 SNEAMN=SIN(EAMN*CFDTR)
      DO 201  IPP=1,NRPP
        RE=RE0+EELV(IPP)/1000.0
        COSELA=COS(ELAT(IPP)*CFDTR)
        XE(IPP)=RE*COSELA*COS(ELON(IPP)*CFDTR)
        YE(IPP)=RE*COSELA*SIN(ELON(IPP)*CFDTR)
```

```
      ZE(IPP)=RE*SIN(ELAT(IPP)*CFDTR)
  201 CONTINUE
  210 DO 212  ISL=1,360
        SLONI=ISL-180
        XS=RGO*COS(SLONI*CFDTR)
        YS=RGO*SIN(SLONI*CFDTR)
        DO 211  IPP=1,NRPP
          XSE=XS-XE(IPP)
          YSE=YS-YE(IPP)
          ZSE=  -ZE(IPP)
          SE=SQRT(XSE*XSE+YSE*YSE+ZSE*ZSE)
          SP=XSE*XE(IPP)+YSE*YE(IPP)+ZSE*ZE(IPP)
          SNEA=SP/(SE*RE)
          IF (IPP.EQ.1)  THEN
            SNEAI=SNEA
          ELSE
            SNEAI=MIN(SNEA,SNEAI)
          END IF
  211   CONTINUE
        IF (SNEAI.LT.SNEAMN)  THEN
          ICALC(ISL)=0
        ELSE
          ICALC(ISL)=ISL
        END IF
  212 CONTINUE
      DO 213  ISL=361,540
        IF (ICALC(ISL-360).GT.0)  THEN
          ICALC(ISL)=ISL
        ELSE
          ICALC(ISL)=0
        END IF
  213 CONTINUE
C Resetting of the ICALC flags for the satellite longitude
C positions other than those for a full continuous range of
C longitude positions  (Statement Nos. 220 - 224+1)
  220 IF (ICALC(1).GT.0)  THEN
        DO 221  ISL=1,180
          IF (ICALC(ISL).GT.0)  THEN
            ICALC(ISL)=0
          ELSE
            GO TO 222
          END IF
  221   CONTINUE
  222   ISLMX=ISL+360
        DO 223  ISL=ISLMX,540
          ICALC(ISL)=0
  223   CONTINUE
      ELSE
        DO 224  ISL=361,540
          ICALC(ISL)=0
  224   CONTINUE
      END IF
C Selection, and storing in the SLON array, of necessary
C satellite longitude positions in orbital arcs where calculation
```

```
C of every 4 degrees will do  (Statement Nos. 230 - 232)
  230 JSL=0
      DO 231  ISL=1,539
        IF (ICALC(ISL).GT.0)  THEN
          IF (JSL.EQ.0)  THEN
            JSL=1
            SLON(JSL)=ISL-180
            ISL0=ISL
          ELSE IF ((ISL-ISL0).EQ.4.OR.ICALC(ISL+1).EQ.0)  THEN
            JSL=JSL+1
            SLON(JSL)=ISL-180
            ISL0=ISL
          END IF
        ELSE
          IF (JSL.GT.0)   GO TO 232
        END IF
  231 CONTINUE
  232 NRSL=JSL
C Calculation and writing of the beam data for the selected
C satellite longitude positions  (Statement Nos. 300 - 329)
  300 IPAGE=IPAGE+1
      IF (IPAGE.EQ.1)  THEN
        WRITE (6,6300)  EAMN,BWMN,PTER,OAER
      ELSE
        WRITE (6,6301)  EAMN,BWMN,PTER,OAER
      END IF
      WRITE (6,'(1X/1X,A8,4X,5HNRSL=,I3)')  NMSAP,NRSL
      DO 319  JSL=1,NRSL
        CALL MNEBSA(EAMN,BWMN,PTER,OAER,
     1              SLON(JSL),NRPP,ELON,ELAT,EELV,
     2              ELVA(JSL),ALON(JSL),ALAT(JSL),
     3              BWMA(JSL),BWMI(JSL),ORAN(JSL),AREA)
        WRITE (6,'(1X,7F8.2)')
     1         SLON(JSL),ELVA(JSL),ALON(JSL),ALAT(JSL),
     2         BWMA(JSL),BWMI(JSL),ORAN(JSL)
  319 CONTINUE
  320 NRSA=NRSA+1
      NMSA(NRSA)=NMSAP
      DO 321  JSL=1,NRSL
        INSO(JSL)=(NINT(SLON(JSL))+180)*200+NINT(ORAN(JSL))
  321 CONTINUE
  329 WRITE (2,REC=NRSA+1)
     1       NMSA(NRSA),NRSL,INSO,ELVA,ALON,ALAT,BWMA,BWMI
      GO TO 110
  399 CONTINUE
C Rewrites the first record of the DFMNEB data file.
  400 WRITE (2,REC=1)
     1       NMDFMN,NRSA,EAMN,BWMN,PTER,OAER
C Writes the names of all service areas as the last file(s) of
C the DFMNEB data file.
  410 IF (NRSA.GT.0)  THEN
        NRRECS=(NRSA+127)/128
        DO 419  IRECS=1,NRRECS
          IREC=1+NRSA+IRECS
```

```
            ISAMN=(IREC-1)*128+1
            ISAMX=MIN(ISAMN+127,NRSA)
            WRITE (2,REC=IREC)
     1           (NMSA(ISA),ISA=ISAMN,ISAMX)
  419    CONTINUE
         END IF
C NORMAL STOP
  800 STOP
C Error Stop
  910 WRITE (6,6910)  IOS
         STOP
  911 WRITE (6,6911)
         STOP
  912 WRITE (6,6912)  IOS
         STOP
  921 WRITE (6,6921)
         STOP
  922 WRITE (6,6922)  IOS
         STOP
  991 WRITE (6,6991)
         STOP
  992 WRITE (6,6992)  IOS
         STOP
C FORMAT Statements
 6300 FORMAT (' CDMNEB    EAMN-',F4.1,'   BWMN-',F4.1,
     1  '   PTER-',F4.1,'   OAER-',F4.1)
 6301 FORMAT ('1CDMNEB    EAMN-',F4.1,'   BWMN-',F4.1,
     1  '   PTER-',F4.1,'   OAER-',F4.1)
 6910 FORMAT (1X//
     1  ' ***   ERROR',I6,'  IN OPENING FILE  DFPLGN   ***'/)
 6911 FORMAT (1X/
     1  ' ***   UNEXPECTED END OF FILE IN READING DFPLGN   ***'/)
 6912 FORMAT (1X//
     1  ' ***   ERROR',I6,'  IN READING FILE  DFPLGN   ***'/)
 6921 FORMAT (1X/
     1  ' ***   UNEXPECTED END OF FILE IN READING DFMNEB   ***'/)
 6922 FORMAT (1X//
     1  ' ***   ERROR',I6,'  IN READING FILE  DFMNEB   ***'/)
 6991 FORMAT (1X/
     1  ' ***   UNEXPECTED END OF FILE IN READING ',
     2  'THE SCRATCH FILE   ***'/)
 6992 FORMAT (1X//
     1  ' ***   ERROR',I6,'  IN READING THE SCRATCH FILE   ***'/)
      END
```

APPENDIX C. THE MNEBIP SUBROUTINE

The MNEBIP subroutine calculates the data (or beam parameters) of elliptical beams of satellite antennas by interpolation. It calculates the beam parameters of an elliptical beam of a satellite antenna for a satellite longitude position for each service area by linearly interpolating the beam parameters stored in an elliptical beam data file, created or appended by the CDMNEB program. The MNEBIP subroutine opens and closes the DFMNEB data file each time the subroutine is called; the user of this subroutine is not required to open or close the data file.

This subroutine is written in ANSI (American National Standards Institute) Standard Fortran (Publication X3.9-1978, ANSI, 345 East 47th Street, New York, NY 10017).

A Fortran listing of the subroutine follows. The user information of the subroutine, including the description of the input and output arguments as well as the input data file, is given in the beginning of the subroutine.

```
      SUBROUTINE  MNEBIP(LUNR,NMDF,NRSA,NMSA,SLON,
     1                   ELVA,ALON,ALAT,BWMA,BWMI,ORAN)
C
C This subroutine calculates the data of elliptical beams of
C satellite antennas by interpolation.
C
C U.S. Department of Commerce, NTIA/ITS.
C Version of 87/12/31.
C
C This subroutine calculates the data of an elliptical beam of a
C satellite antenna for a satellite longitude position for each
C service area by interpolating the data stored in an elliptical
C beam data file, created or appended by the CDMNEB program.
C
C This subroutine does, for each service area,
C (1) retrieve, from the data file, the parameters of
C     minimum elliptical beams for all satellite longitude
C     positions for the service area,
C (2) select two satellite longitude positions that bound,
C     or are closest to, the given satellite longitude
C     position for which calculation of elliptical beam is
C     desired, and
C (3) linearly interpolate the desired parameters from the
C     data for the selected two satellite longitude
C     positions.
C
C The input arguments are
C   LUNR   - logical unit number of the data file,
C   NMDF   - character string of length six (6) for name of
C            the data file,
C   NRSA   - number of service areas,
C   NMSA   - character array of dimension NRSA and of length
C            eight (8) containing the names of the service
C            areas,
C   SLON   - array of dimension NRSA containing the satellite
C            longitudes (in degrees) for which calculations
C            of elliptical beam data are desired.
C
C The output arguments are all arrays of dimension NRSA, where
C the following variables are to be stored:
C   ELVA   - elevation angle of the satellite for the service
C            area (in degrees), defined as the minimum value
C            of the elevation angle of the satellite seen
C            from the earth points of the service area,
C   ALON   - longitude of the aim point (or boresight) of the
C            elliptical beam of the satellite antenna on the
C            surface of the Earth (in degrees),
C   ALAT   - latitude of the aim point (or boresight) of the
C            elliptical beam of the satellite antenna on the
C            surface of the Earth (in degrees),
C   BWMA   - major-axis beamwidth of the elliptical beam
C            (in degrees),
C   BWMI   - minor-axis beamwidth of the elliptical beam
C            (in degrees),
```

```
C   ORAN   - orientation angle of the ellipse (in degrees),
C            i.e., in a plane normal to the beam axis, the
C            angle measured counterclockwise from a line
C            parallel to the equatorial plane to the major
C            axis of the ellipse to the nearest degree.
C
C This subroutine opens, as UNIT-LUNR and FILE-NMDF, a direct-
C access, unformatted data file with the record length set to
C 1024 bytes.  This file contains the data of minimum elliptical
C beam of satellite antenna for the service areas and for
C satellite longitude positions.
C
C The Data File Contents -
C   The first record of the file contains the following data:
C     (1) the file name,
C     (2) NRSA, i.e., the total number of service areas
C         included in this file,
C     (3) EAMN, i.e., the minimum elevation angle of the
C         satellite for the service area (in degrees) for
C         which this program calculates the minimum
C         elliptical beam,
C     (4) BWMN, i.e., the minimum beamwidth (in degrees),
C     (5) PTER, i.e., the pointing error (in degrees), and
C     (6) OAER, i.e., the orientation-angle error
C         (in degrees).
C   Each of the following NRSA records contains the following
C   data for each service area:
C     (1) NMSA, i.e., the name of the service area,
C     (2) NRSL,  i.e., number of satellite longitude
C         positions,
C     (3) SLON, i.e., the satellite longitudes (in degrees),
C     (4) ELVA, i.e., the elevation angles of the satellite
C         for the service area (in degrees), defined as the
C         minimum value of the elevation angle of the
C         satellite seen from the earth points of the
C         service area,
C     (5) ALON, i.e., the longitudes of the aim points
C         (or boresight) of the elliptical beams of the
C         satellite antenna on the surface of the Earth
C         (in degrees),
C     (6) ALAT, i.e., the latitudes of the aim points
C         (or boresight) of the elliptical beams of the
C         satellite antenna on the surface of the Earth
C         (in degrees),
C     (7) BWMA, i.e., the major-axis beamwidths of the
C         elliptical beams (in degrees),
C     (8) BWMI, i.e., the minor-axis beamwidths of the
C         elliptical beams (in degrees), and
C     (9) ORAN, i.e., the orientation angles of the ellipse
C         (in degrees), each defined in a plane normal to
C         the beam axis as the angle measured counterclock-
C         wise from a line parallel to the equatorial plane
C         to the major axis of the ellipse to the nearest
C         degree.
```

```
C   The last (NRSA+127)/128 file(s) contain(s) the names of all
C   NRSA service areas.
C
C Specification Statements
C - PARAMETER Statements
      PARAMETER   (NRSAMX=500)
      PARAMETER   (NRSLMX=42)
C - DIMENSION Statements
      DIMENSION   NMSA(*),SLON(*),
     1            ELVA(*),ALON(*),ALAT(*),BWMA(*),BWMI(*),ORAN(*)
      DIMENSION   NMSAD(NRSAMX)
      DIMENSION   SLONA(NRSLMX),ELVAA(NRSLMX),
     1            ALONA(NRSLMX),ALATA(NRSLMX),
     2            BWMAA(NRSLMX),BWMIA(NRSLMX),
     3            ORANA(NRSLMX),INSOA(NRSLMX)
C - TYPE Statements
      CHARACTER*6 NMDF
      CHARACTER*8 NMDMMY,NMSA,NMSAD
C
C Calculation
C Initialization
   10 OPEN  (UNIT=LUNR,FILE=NMDF,STATUS='OLD',ERR=900,IOSTAT=IOS,
     1       ACCESS='DIRECT',FORM='UNFORMATTED',RECL=1024)
      READ  (LUNR,REC=1,END=910,ERR=920,IOSTAT=IOS)
     1       NMDMMY,NRSAD
      IF (NRSAD.LE.0)     GO TO 910
   20 NRRECS=(NRSAD+127)/128
      DO 21  IRECS=1,NRRECS
        IREC=1+NRSAD+IRECS
        ISADMN=(IRECS-1)*128+1
        ISADMX=MIN(ISADMN+127,NRSAD)
        READ  (LUNR,REC=IREC,END=910,ERR=920,IOSTAT=IOS)
     1        (NMSAD(ISAD),ISAD=ISADMN,ISADMX)
   21 CONTINUE
   30 DO 31  ISA=1,NRSA
        ELVA(ISA)=-100.0
   31 CONTINUE
   40 NRSAI=0
C Outermost DO-loop with respect to the service area, NMSAD, in
C the DFMNEB data file
  100 DO 490  ISAD=1,NRSAD
        DO 110  ISA=1,NRSA
          IF (NMSA(ISA).EQ.NMSAD(ISAD))  GO TO 200
  110   CONTINUE
        GO TO 490
C Retrieval of the data for the desired service area
  200   READ  (LUNR,REC=ISAD+1,END=910,ERR=920,IOSTAT=IOS)
     1         NMSAD(ISAD),NRSL,
     2         INSOA,ELVAA,ALONA,ALATA,BWMAA,BWMIA
        DO 201  ISL=1,NRSL
          SLONA(ISL)=INSOA(ISL)/200-180
          ORANA(ISL)=MOD(INSOA(ISL),200)
  201 CONTINUE
C Selection of two satellite longitude positions in the data file
```

```
C that enclose the desired position
  300   SLONI=MOD(SLON(ISA)+900.0,360.0)-180.0
        IF (SLONI.LT.SLONA(1))      SLONI=SLONI+360.0
        IF (SLONI.LT.SLONA(1).OR.SLONI.GT.SLONA(NRSL))
     1                                GO TO 440
        IF (SLONI.EQ.SLONA(1))  THEN
          ISL1=1
          GO TO 420
        END IF
        DO 310  ISL=2,NRSL
          IF (SLONI.LT.SLONA(ISL))  THEN
            ISL1=ISL-1
            ISL2=ISL
            GO TO 400
          ELSE IF (SLONI.EQ.SLONA(ISL))  THEN
            ISL1=ISL
            GO TO 420
          END IF
  310   CONTINUE
C Linear interpolation of data between the two selected satellite
C longitude positions
  400   RSLON=(SLONI-SLONA(ISL1))/(SLONA(ISL2)-SLONA(ISL1))
        ELVA(ISA)=ELVAA(ISL1)+(ELVAA(ISL2)-ELVAA(ISL1))*RSLON
        ALON1=MOD(ALONA(ISL1)+900.0,360.0)-180.0
        ALON2=MOD(ALONA(ISL2)+900.0,360.0)-180.0
        DALON=ALON2-ALON1
        IF (DALON.LT.-180.0)  THEN
          DALON=DALON+360.0
        ELSE IF (DALON.GT.180.0)  THEN
          DALON=DALON-360.0
        END IF
        ALONI=ALON1+DALON*RSLON
        ALONI=ANINT(ALONI*10.0)/10.0
        ALON(ISA)=MOD(ALONI+900.0,360.0)-180.0
        ALATI=ALATA(ISL1)+(ALATA(ISL2)-ALATA(ISL1))*RSLON
        ALAT(ISA)=ANINT(ALATI*10.0)/10.0
        BWMAI=BWMAA(ISL1)+(BWMAA(ISL2)-BWMAA(ISL1))*RSLON
        BWMA(ISA)=AINT(BWMAI*100.0)/100.0
        IF (BWMAI.GT.BWMA(ISA))      BWMA(ISA)=BWMA(ISA)+0.01
        BWMII=BWMIA(ISL1)+(BWMIA(ISL2)-BWMIA(ISL1))*RSLON
        BWMI(ISA)=AINT(BWMII*100.0)/100.0
        IF (BWMII.GT.BWMI(ISA))      BWMI(ISA)=BWMI(ISA)+0.01
        ORAN1=MOD(ORANA(ISL1)+900.0,180.0)
        ORAN2=MOD(ORANA(ISL2)+900.0,180.0)
        IF (BWMAA(ISL1).LE.BWMIA(ISL1))  THEN
          ORAN(ISA)=ORAN2
        ELSE IF (BWMAA(ISL2).LE.BWMIA(ISL2))  THEN
          ORAN(ISA)=ORAN1
        ELSE
          DORAN=ORAN2-ORAN1
          IF (DORAN.LT.-90.0)  THEN
            DORAN=DORAN+180.0
          ELSE IF (DORAN.GT.90.0)  THEN
            DORAN=DORAN-180.0
```

```
            END IF
            ORANI=ORAN1+DORAN*RSLON
            ORANI=ANINT(ORANI)
            ORAN(ISA)=MOD(ORANI+900.0,180.0)
          END IF
          GO TO 480
C Special Case -- No interpolation (SLON found in the NMDF file)
  420     ELVA(ISA)=ELVAA(ISL1)
          ALON(ISA)=ALONA(ISL1)
          ALAT(ISA)=ALATA(ISL1)
          BWMA(ISA)=BWMAA(ISL1)
          BWMI(ISA)=BWMIA(ISL1)
          ORAN(ISA)=ORANA(ISL1)
          GO TO 480
C Special Case -- Invisible orbital position
  440     ELVA(ISA)=-90.0
          ALON(ISA)=0.0
          ALAT(ISA)=0.0
          BWMA(ISA)=0.0
          BWMI(ISA)=0.0
          ORAN(ISA)=0.0
  480     NRSAI=NRSAI+1
          IF (NRSAI.EQ.NRSA)      GO TO 500
  490   CONTINUE
        GO TO 930
C Normal Return
  500   CLOSE (LUNR)
        RETURN
C Error Stop
  900   WRITE (6,6900)  IOS
        STOP
  910   WRITE (6,6910)
        STOP
  920   WRITE (6,6920)  IOS
        STOP
  930   WRITE (6,6930)
        DO 931  ISA=1,NRSA
          IF (ELVA(ISA).LT.-90.0)
     1      WRITE (6,'(16X,A8)')  NMSA(ISA)
  931   CONTINUE
        STOP
C FORMAT Statements
 6900   FORMAT (1X//
     1    ' ***   ERROR',I6,'  IN OPENING FILE  DFMNEB   ***'/)
 6910   FORMAT (1X/
     1    ' ***   DATA FILE ERROR  --  DFMNEB IS EMPTY   ***'/)
 6920   FORMAT (1X//
     1    ' ***   ERROR',I6,'  IN READING FILE  DFMNEB   ***'/)
 6930   FORMAT (1X/
     1    ' ***   DATA FOR NMSA NOT FOUND IN DFMNEB   ***'/)
        END
```

FORM NTIA-29
(4-80)

U.S. DEPARTMENT OF COMMERCE
NAT'L. TELECOMMUNICATIONS AND INFORMATION ADMINISTRATION

BIBLIOGRAPHIC DATA SHEET

	1. PUBLICATION NO.	2. Gov't Accession No.	3. Recipient's Accession No.
	NTIA Report 88-230		

4. TITLE AND SUBTITLE

Use of Precalculated Data of Minimum Elliptical Beams of Satellite Antennas

5. Publication Date
February 1988

6. Performing Organization Code
ITS.N1

7. AUTHOR(S)
Hiroshi Akima

8. PERFORMING ORGANIZATION NAME AND ADDRESS
National Telecommunications and Information Admin.
Institute for Telecommunication Sciences
325 Broadway
Boulder, CO 80303

9. Project/Task/Work Unit No.
9108142

10. Contract/Grant No.

11. Sponsoring Organization Name and Address
National Telecommunications and Information Admin.
Herbert C. Hoover Building
14th & Constitution Ave., N.W.
Washington, D.C. 20230

12. Type of Report and Period Covered

13.

14. SUPPLEMENTARY NOTES

15. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

Minimum (or "best-fit") elliptical beams of satellite antennas have been used for the planning of space services utilizing the geostationary-satellite orbit. In such a planning, beam parameters of minimum elliptical beams are necessary for all possible orbital positions of all satellites. Since calculation of beam parameters of every minimum elliptical beam may be rather time-consuming, an alternative method is desirable. This study shows that the beam parameters for a satellite longitude position calculated by linear interpolation from the beam parameters for two nearby longitude positions can be used without a serious degradation of antenna gain. This report describes a method in which the beam parameters for minimum elliptical beams are precalculated for a sufficient number of satellite longitude positions and the beam parameters for a desired longitude position are calculated from the beam parameters for two nearby longitude positions that bound the longitude position in question. Fortran programs that implement this method also are presented.

16. Key Words *(Alphabetical order, separated by semicolons)*

antenna beam; elliptical beam; linear interpolation; minimum elliptical beam; satellite antenna

17. AVAILABILITY STATEMENT

☒ UNLIMITED.

☐ FOR OFFICIAL DISTRIBUTION.

18. Security Class *(This report)*
None

20. Number of pages
60

19. Security Class *(This page)*
None

21. Price

U.S. GOVERNMENT PRINTING OFFICE: 1988—573-002/80,517 REGION NO. 8

NTIA FORMAL PUBLICATION SERIES

NTIA MONOGRAPH

A scholarly, professionally oriented publication dealing with state-of-the-art research or an authoritative treatment of a broad area. A monograph is expected to have a long lifespan.

NTIA SPECIAL PUBLICATION

Information derived from or of value to NTIA activities such as conference proceedings, bibliographies, selected speeches, course and instructional materials, and directories.

NTIA HANDBOOK

Information pertaining to technical procedures; reference and data guides, and formal user's manuals that are expected to be pertinent for a long time.

NTIA REPORT

Important contributions to existing knowledge but of less breadth than a monograph, such as results of completed projects and major activities, specific major accomplishments, or NTIA-coordinated activities.

NTIA RESTRICTED REPORT

Contributions that fit the NTIA Report classification but that are limited in distribution because of national security classification or Departmental constraints. This material receives full review and quality control equivalent to the open-literature report series.

NTIA CONTRACTOR REPORT

Information generated under an NTIA contract or grant and considered an important contribution to existing knowledge.

SPONSOR-ISSUED REPORTS

NTIA authors occasionally produce reports issued under an other-agency sponsor's cover. These reports generally embody the criteria of the NTIA Report series.

For information about NTIA publications, contact the Executive Office at 325 Broadway, Boulder, Colorado 80303 (telephone: 303-497-3572).

This report is for sale by the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.